(19)日本国特許庁（ＪＰ）　(12)**公開特許公報（Ａ）**　(11)特許出願公開番号

特開平8－204697

(43)公開日　平成８年(1996)８月９日

(51)Int.Cl.[6]		識別記号	庁内整理番号	ＦＩ	技術表示箇所
Ｈ０４Ｌ	9/00				
	9/10				
	9/12				
Ｇ０９Ｃ	1/00		7259－5Ｊ		
				Ｈ０４Ｌ　9/00　Ｚ	

審査請求　未請求　請求項の数７　ＯＬ　（全 24 頁）

(21)出願番号　特願平7－8185

(22)出願日　平成７年(1995)１月23日

(71)出願人　000001007
キヤノン株式会社
東京都大田区下丸子３丁目30番２号

(72)発明者　マヌエル・セレセド
東京都大田区下丸子３丁目30番２号キヤノン株式会社内

(72)発明者　岩村　恵市
東京都大田区下丸子３丁目30番２号キヤノン株式会社内

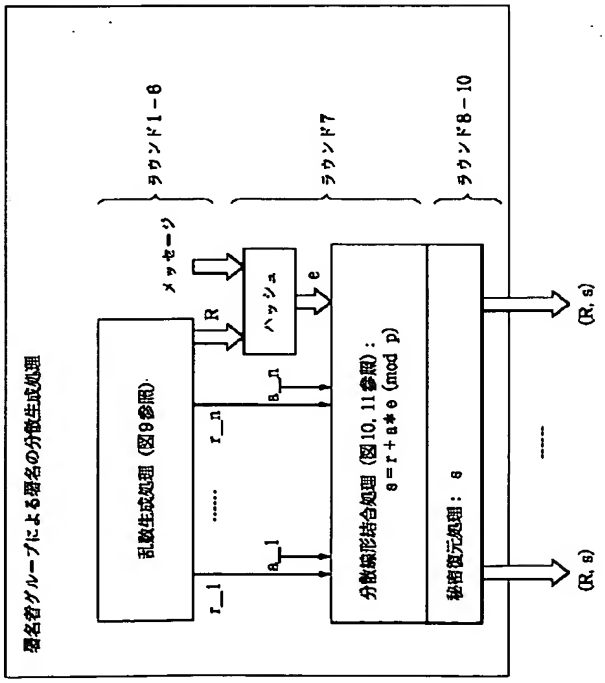
(74)代理人　弁理士　丸島　儀一

(54)【発明の名称】　複数の装置を有する通信システムにおける署名生成方法

(57)【要約】

【目的】　実用的な計算量と通信量とで分散して署名の生成を可能とする。

【構成】　複数の装置が、個別の装置間で他の装置には秘密に情報の通信を行なうための秘密通信路と、各装置から他の全ての装置へ共通に情報を送信するための放送通信路とを介して接続された通信システムにおいて、署名者グループ内の各装置が、第１の秘密情報をランダムに選び、前記グループ内の各装置に秘密に分散させ、各装置が、前記第１の秘密情報に所定の第１の関数を作用させ、得られた出力値を前記グループ内の全装置に放送し、各装置の前記第１の秘密情報を分散加算し、各装置の前記出力値を分散乗算し、当該乗算の結果とメッセージとを所定の第２の関数で処理し、該処理結果と、前記分散加算の結果と、公開された元とを用いて第２の秘密情報を分散演算し、分散された第２の秘密情報を復元し、前記分散乗算結果と共に署名として出力することを特徴とする。



【特許請求の範囲】
【請求項１】　複数の装置が、個別の装置間で他の装置には秘密に情報の通信を行なうための秘密通信路と、各装置から他の全ての装置へ共通に情報を送信するための放送通信路とを介して接続された通信システムにおいて、
署名者グループ内の各装置が、第１の秘密情報をランダムに選び、前記グループ内の各装置に秘密に分散させ、
各装置が、前記第１の秘密情報に所定の第１の関数を作用させ、得られた出力値を前記グループ内の全装置に放送し、
各装置の前記第１の秘密情報を分散加算し、
各装置の前記出力値を分散乗算し、当該乗算の結果とメッセージとを所定の第２の関数で処理し、
該処理結果と、前記分散加算の結果と、公開された元とを用いて第２の秘密情報を分散演算し、
分散された第２の秘密情報を復元し、前記分散乗算結果と共に署名として出力することを特徴とする署名生成方法。
【請求項２】　前記第２の秘密情報の分散演算が線形結合であることを特徴とする請求項１に記載の署名生成方法。
【請求項３】　前記第２の秘密情報の分散演算がべき乗を含む積による結合であることを特徴とする請求項１に記載の署名生成方法。
【請求項４】　前記第２の秘密情報の分散演算が線形結合と乗算との組み合わせであることを特徴とする請求項１に記載の署名生成方法。
【請求項５】　前記第２の関数が一方向性の関数であることを特徴とする請求項１に記載の署名生成方法。
【請求項６】　前記第１の秘密情報は所定の有限体上の元であり、各演算は該有限体上の演算を利用することを特徴とする請求項１に記載の署名生成方法。
【請求項７】　前記第１の秘密情報の分散を、
該第１の秘密情報を有する第１の装置が、当該秘密情報から所定の部分配列を生成し、
前記第１の装置が、前記部分配列より他の装置の各々に対する第１の部分情報を夫々抽出して、各装置に前記秘密通信路を介して送信し、
前記第１の装置が、前記第１の部分情報に所定の関数を作用させ、得られた出力値を、各装置に前記放送通信路を介して放送し、
前記各装置が、乱数を生成し、生成された乱数を前記放送通信路を介して放送し、
前記第１の装置が、放送された前記乱数の値に応じて、前記部分配列より第２の部分情報を生成し、生成された第２の部分情報を前記放送通信路を介して放送し、
前記各装置が、前記第１の部分情報及び前記生成された乱数に応じて、前記第１の情報処理装置で第２の部分情報として生成されるべき第３の部分情報を生成し、
前記各装置が、前記第３の部分情報と放送された第２の部分情報とを比較して、前記第１の装置による秘密の分散を確認することを特徴とする請求項１に記載の署名生成方法。
【発明の詳細な説明】
【０００１】
【産業上の利用分野】本発明は、通信路によって接続された複数の加入者の内の何人かの加入者によるグループにおけるディジタル署名を、そのグループに参加する加入者間で分散生成する方式に関するものである。
【０００２】
【従来の技術】通信路によって接続された複数の情報処理装置を含む情報通信システムにおいて、送信された情報が指定された受信装置以外に漏れないこと（情報の秘匿）を保証するために役立つ技術の一つとして、暗号技術が知られている（池野、小山：”現代暗号理論”、pp. 224-225 、電子情報通信学会）。
【０００３】暗号技術は、上述した情報の秘匿機能の他に、その通信システムにおいて受け取った情報が、指示された装置から発信されたこと（途中で改竄されなかったこと）を確認できる認証機能、及びその受け取った情報が指示された装置から発信されたことを第三者にも証明できるディジタル署名と呼ばれる機能を実現するために有効であることもよく知られている。
【０００４】特に、公開鍵暗号方式の１つであるＲＳＡ暗号を用いた認証及びディジタル署名方式は広く用いられている（例えば、R. Rivest, A. Shamir, L. Adleman, ”AMethod for Obtaining Digital Signatures and Public-Key Cryptosystems”, Communications of the ACM, 21, 2, 1978, pp. 120-126. または米国特許4,405,828 号参照）。
【０００５】また、ＲＳＡ暗号以外の公開鍵暗号方式によるディジタル署名方式の１つとして、A. Fiat, A. Shamirによって提案された方式がよく知られている（”How toProve Yourself: Practical Solutions to Identification and Signature Problems”, Advances in Cryptology —Crypto'87, Lecture Notes in Computer Science, 263, Springer-Verlag, 1988, pp. 186-194. または米国特許 4,748,668号参照）。この方式では、安全な識別及び与えられたメッセージのディジタル署名を実現するために次の処理を行う。
【０００６】(1) 与えられたメッセージのディジタル署名を計算する装置、あるいは通信システムにおいて信頼できるセンタの役割を果たす装置が、1, ･･･, N-1 （但しN は２つの素数p, q の積である）の内から元a をランダムに選び、署名を計算する装置の秘密情報とする。
【０００７】(2) その秘密a を選んだ装置は、$a^{l} \bmod N$ （但し、l はgcd(l, lambda(N))=1を満たし、ここで、lambda(N)=lcm(p-1, q-1)、gcd(a, b)はa, b の最大公約数、lcm(a, b)はa, b の最小公倍数を意味する）を計算し、そ

の装置によって生成されたディジタル署名を確認するための公開情報とする。

【０００８】(3) その装置によって与えられたメッセージm のディジタル署名の生成処理において、 {1,･･･,N-1 } の内からランダムに選ばれた秘密の元r を用いて計算されたR=r^l mod N と、与えられた公開のメッセージm を連続させた値R|m を入力として所定の関数h を用いてe=h(R|m)を計算する。それらを入力としてs=r*a^e(mod N) の計算を実行することによって得られた出力s 及びR を、与えられたメッセージのディジタル署名とする。

【０００９】(4) ある与えられたメッセージm に対するディジタル署名(s,R) を確認するためには、s^l mod N とR*(a^l)^(h(R|m)) mod Nの計算を実行し、それらの結果が等しいことを確認する。

【００１０】また、C.P.Schnorr によって提案された安全な識別及び与えられたメッセージのディジタル署名を実現する方式（"Efficient Identification and Signaturesfor Smart Cards", Advances in Cryptology —Crypto'89, Lecture Notes inComputer Science, 435, Springer-Verlag, 1990, pp. 239-252. または米国特許4,995,082 号参照）は、次の処理を行う。

【００１１】(1) 与えられたメッセージのディジタル署名を計算する装置、あるいは通信システムにおいて信頼できるセンタの役割を果たす装置は {1,･･･,p } （但しp は素数である）の内から元a をランダムに選び、署名を計算する装置の秘密情報とする。

【００１２】(2) その秘密a を選んだ装置は、g^{-a} mod q （但し、q はp がq-1 の除数になるような素数であり、有限体 GF(q)に属する元g の位数がp となる）を計算し、その装置によって生成されたディジタル署名を確認するための公開情報とする。

【００１３】(3) その装置によって与えられたメッセージm のディジタル署名の生成処理において、 {1,･･･,p } の内からランダムに選ばれた秘密の元r を用いて計算された R=g^r mod q と、与えられた公開のメッセージm を連続させた値R|m を入力として所定の関数h を用いてe=h(R|m)を計算する。それらを入力としてs=r+a*e (mod p) の計算を実行することによって得られた出力s 及びR を、与えられたメッセージのディジタル署名とする。

【００１４】(4) ある与えられたメッセージm に対するディジタル署名(s,R) を確認するためには、h(g^s((g^(-a))^(h(R|m))) (mod q)|m)R|m))の計算（但しx^y はx yを表す）を実行し、この結果とe とが等しいことを確認する。

【００１５】また、T. ElGamalによって提案された安全な識別及び与えられたメッセージのディジタル署名を実現する方式（"A Public-Key Cryptosystem and a SignatureScheme Based on Discrete Logarithms", IEEE Transactions on InformationTheory, IT-31, 4, 1985, pp. 469-472. American National Standard X9.30-199x, Digital Signature Algorithm, Feb. 1992参照）は、次の処理を行う。

【００１６】(1) 与えられたメッセージのディジタル署名を計算する装置、あるいは通信システムにおいて信頼できるセンタの役割を果たす装置、は {1,･･･,p } （但しpは素数である）の内から元a をランダムに選び、署名を計算する装置の秘密情報とする。

【００１７】(2) その秘密a を選んだ装置は、g^(-a) mod q （但し、q はp がq-1 の除数になるような素数であり、 GF(q)に属する元g の位数がp となる）を計算し、その装置によって生成されたディジタル署名を確認するための公開情報とする。

【００１８】(3) その装置によって与えられたメッセージm のディジタル署名の生成処理において、 {1,･･･,p } の内からランダムに選ばれた秘密の元r を用いて計算された R=g^r mod q と、所定の関数h を用いて与えられた公開のメッセージm を入力として得られた値e=h(m)とを入力として、s=(e+R*a)*r^{-1} mod p の計算を実行することによって得られた出力s 、及びR を与えられたメッセージのディジタル署名とする。

【００１９】(4) ある与えられたメッセージm に対するディジタル署名(s,R) を確認するためには、(g^(-a))^R (g^r)^s とg^m (mod q) の計算を実行し、それらの結果が等しいことを確認する。

【００２０】一方、前述のように守秘及び認証を実現する情報通信システムにおいて、秘密情報を守りながら信頼性を高める手段として、秘密情報を分散し、与えられたメッセージのディジタル署名の計算を通信路によって接続された複数の計算装置（以下、署名者グループといい、そのグループに参加する各計算装置を加入者という、またそのグループに参加する加入者の数をn で表す）の間で分散する方法がY. Desmedt,Y. Frankelによって提案されている（"Threshold Cryptosystems",Advances in Cryptology—Crypto'89, 435, Springer-Verlag, 1990, pp. 307-315: "Shared Generation of Authenticators and Signatures", Advances in Cryptology—Crypto'91, 576, Springer-Verlag, 1992, pp. 457-469参照）。

【００２１】この分散ディジタル署名方式の基本的な部分は、複数の加入者からなる通信システムにおいて秘密情報を分散する方式である。具体的に、ある秘密情報x が複数の加入者間で分散保持されるとは、以下の条件(a), (b)が満たされるように各加入者i が秘密情報x に対応する部分情報x_i を生成し、他の加入者に分配することを意味する。

【００２２】(a) 秘密情報x を復元するためにt+1 人の加入者の部分情報が必要である。以後、その秘密情報を復元するための必要な加入者の数t+1 をしきい値と呼

ぶ。

【００２３】(b) しきい値未満の数（t 以下）の部分情報ではその秘密情報に関するどのような情報も得ることができない。

【００２４】従来の基本的な秘密分散方式は、A. Shamir によって提案され"How to Sharea Secret", Communications of the ACM,Vol. 22, 11, 1979参照)、次のように実現された。すなわち、ある加入者の情報を秘密に複数の加入者に分散するために、定数項が前述の秘密情報となるｔ次の多項式f(x)をランダムに選び、n 個の異なる値に対するその多項式の値f(i)(i=1,･･･,n) を各加入者に配る。この加入者i に配られる多項式の値f(i)が前述の部分情報の一部分になる。よって、秘密情報はt+1 個の部分情報を用いた多項式補間によって復元できる（t 以下の部分情報では秘密情報に関するどのような情報を得ることはできない）。

【００２５】前述のY. Desmedt, Y. Frankelによる秘密分散方式に基づくＲＳＡ暗号を用いた分散型ディジタル署名方式は以下の条件(I),(II)を満たす。

【００２６】(I) 署名者グループに対する与えられたメッセージのディジタル署名を生成するためにt+1 人の加入者の協力があれば十分である。

【００２７】(II)しきい値未満の数（t 以下）の加入者では与えられたメッセージのディジタル署名を生成できない。

【００２８】但し、(I),(II)の条件だけではディジタル署名生成処理が分散されたとき、t+1 以上の加入者が協力しても不正な加入者があった場合には、署名を生成できないことがありうる。

【００２９】それに対して、どのような誤りを持つ加入者に対してもディジタル署名を生成できるような方式は、確認可能な秘密分散方式に基づいて構成できることが知られている。どのような誤りを持つ加入者にも耐えられる秘密分散方式として提案された確認可能な秘密分散（Verifiable Secret Sharing ）とは、前述の条件(a), (b)に次の条件(c),(d) を付加することによって定義される。

【００３０】(c) 不正な部分情報と正しい部分情報とが混在していても、t+1 個の正しい部分情報があれば元の秘密情報を復元するために十分である。

【００３１】(d) 全ての加入者がその秘密の部分情報を受け取ったとき、その部分情報がある秘密情報x を復元するために正しい情報であるかどうかを確認できる。

【００３２】前述の条件(c),(d) を満たす確認可能な秘密分散方式に基づいて、次の条件を満たす分散ディジタル署名方式を構成できる。

【００３３】(I')不正な加入者が正しい加入者と混在していても与えられたメッセージのディジタル署名を生成するためにt+1 人の正しい加入者の協力があれば十分である。

【００３４】(II)しきい値未満の数（t 以下）の加入者では与えられたメッセージのディジタル署名を生成できない。

【００３５】秘密の通信路を持つ通信システムに対して、全加入者の３分の１より少ない数であればどのような誤りを持つ加入者にも耐えられる確認可能な秘密分散方式（しきい値ｔがｔ＜ｎ／３を満たす場合）を構成するためには、従来の誤り訂正符号技術で十分であることが、M. Ben-Or, S. Goldwasser, A. Wigdersonによって示されている（"Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation", ACM STOC 1988 参照。）。

【００３６】更に、全加入者の半分より少ない数であれば、どのような誤りを持つ加入者にも耐えられる確認可能な秘密分散方式を構成するためには、条件を更に追加する必要があり、全ての加入者が同じメッセージを受信したことを確認できる放送型通信路を、全ての加入者が持つとした場合、次の２通りの構成方法が知られている。

【００３７】(1) 零知識対話証明システム（辻井、笠原：" 暗号と情報セキュリティ" 、昭晃堂、1990年、参照）で用いられている`Cut and Choose'と呼ばれる技術を利用し、前述のA.Shamirによる基本的な秘密分散方式によって元の秘密s を分散した上に、配られた部分情報s_i （i=1,･･･,n ）を更に分散する方式。つまり、確認可能な分散方式によって配られた全ての部分情報は、秘密部分s_i に対する秘密部分になるように生成された部分行列と考えることができる。

【００３８】但し、前述の`Cut and Choose'技術を用いることで、前述の条件(d) による確認とは統計的確認になり、秘密情報を復元できる正しい分散が行われているかどうかを示す判定出力に誤り確率が生じる。但し、その誤り確率は設定する安全性のパラメータにより無視できる程度まで小さくすることができる。具体例として、T. Rabin, M. Ben-Or による方式("Verifiable Secret Sharing and Multiparty Protocols with Honest Majority", ACM STOC 1989 参照があげられる。

【００３９】(2) 非対話型で特殊な代数学的な性質を持つ一方向性関数を利用する方法。但し、このように構成された秘密分散方式の安全性は、その代数学的な性質を満たす一方向性関数の逆元を計算するのが困難である（実用的な逆元計算方式が存在しない）という暗号的な仮定をする必要が生じる。具体例は、P. Feldmanによって提案された（"A Practical Scheme for Non-Interactive Verifiable SecretSharing", IEEE FOCS, 1987 参照。

【００４０】また、上述の確認可能な秘密分散方式を利用することによって、ディジタル署名の分散生成処理を含めた、ある与えられた有限体上の分散演算を安全に実現する回路を構成できることが T. Rabin, M. Ben-Or

("Verifiable Secret Sharingand Multiparty Protocols with Honest Majority", ACM STOC, 1989 参照)、D. Beaver ("Secure Multiparty Protocols and Zero-Knowledge Proof SystemsTolerating a Faulty Minority", Journal of Cryptology, 1991, 4, pp. 75-122; "Efficient Multiparty Protocols Using Circuit Randomization", Advances in Cryptology-Crypto'91, 1992参照) 及び M. Franklin, S. Haber ("JointEncryption and Message-Efficient Secure Computation", Advances in Cryptology-Crypto'93, 1994 参照) によって示されている。

【００４１】このような確認可能な秘密分散方式を用いた分散演算回路によって、前述のような種々のディジタル署名方式に対する分散生成処理システムも構成できることが知られている。

【００４２】

【発明が解決しようとしている課題】上述の条件(I), (II)を満たす分散ディジタル署名方式に必要な通信量及び計算量は実用的であることが知られている。しかし、前述したように署名の分散生成処理に参加する加入者が不正を行った場合に署名の生成ができないことが有り得る。

【００４３】それに対して、上述の確認可能な秘密分散方式及び分散演算回路を利用する条件(I'), (II) を満たす分散ディジタル署名方式は、以下のように非実用的な通信量または計算量が必要であることが知られている。例えば、対話型確認可能な秘密分散方式(1) において1ビットを分散するために必要な通信量は、安全性のパラメータをk （通常は１００程度の値が用いられる）と表したときn 個の秘密部分に対してn^3k^2のオーダーとなることが知られており、効率的ではない。

【００４４】また、一方向性関数を用い、通信量の少ない非対話型確認可能な秘密分散方式(2) は、n 個の秘密部分に対してn 回のオーダーの特殊な一方向性関数の計算処理を行う必要があり、特に秘密分散処理を安全に分散計算を実行するための部分処理として用いる場合には、実行すべき秘密分散処理の数が多くなり（例えば、分散乗算においてはn^2のオーダーとなる）、全体的に非実用的な計算量になる。

【００４５】以上のように、上述の従来技術による対話型確認可能な秘密分散方式(1) に基づいた分散ディジタル署名は通信量が非常に大きく、非対話型確認可能な秘密分散方式(2) に基づいた分散ディジタル署名は計算量が非常に大きいという問題があった。

【００４６】本発明は、前述の対話的な方式(1) 及び非対話的な方式(2) の中間に位置し、必要な計算量と通信量の両方が実用的なオーダーになる確認可能な秘密分散方式を利用し、条件(I) 、（不正を行う加入者があれば署名を生成できないことが有り得る方式）及び条件(I')（あるしきい値以下の不正を行う加入者の数があっても署名を生成できる方式）の中間に位置する分散ディジタル署名方式（あるしきい値以下の不正を行う加入者の数があった場合、署名を生成できないことが有り得るが、不正を行った加入者は識別できる方式）を提案する。

【００４７】

【課題を解決するための手段】上述の課題を解決するために、本発明の署名生成方法は、複数の装置が、個別の装置間で他の装置には秘密に情報の通信を行なうための秘密通信路と、各装置から他の全ての装置へ共通に情報を送信するための放送通信路とを介して接続された通信システムにおいて、署名者グループ内の各装置が、第１の秘密情報をランダムに選び、前記グループ内の各装置に秘密に分散させ、各装置が、前記第１の秘密情報に所定の第１の関数を作用させ、得られた出力値を前記グループ内の全装置に放送し、各装置の前記第１の秘密情報を分散加算し、各装置の前記出力値を分散乗算し、当該乗算の結果とメッセージとを所定の第２の関数で処理し、該処理結果と、前記分散加算の結果と、公開された元とを用いて第２の秘密情報を分散演算し、分散された第２の秘密情報を復元し、前記分散乗算結果と共に署名として出力することを特徴とする。

【００４８】

【実施例】以下、図面を参照して本発明の実施例を詳細に説明する。

【００４９】本実施例では、各加入者が、他の個々の加入者にメッセージを秘密に送ることができる秘密通信路と、全ての加入者が同じメッセージを受けたことを確認できる放送型通信路とによって接続される分散システムにおいて、図３に示すように、後述する秘密部分行列を生成し、その秘密の部分情報を分配すると同時に、後述する部分情報の認証子として用いられる一方向性ハッシュ値（但し、特殊な代数学的な性質を持つ必要はない）を放送し、後述するCut and Choose処理を行う秘密分散方式を用いて、署名者グループに参加する各加入者によってランダムに選ばれた秘密の元をそのグループの全ての加入者間で分散し、正しく分散されたランダムな秘密の分散演算処理を実行することを特徴する。

【００５０】まず、本実施例で利用する技術及び用語のいくつかについて説明する。

【００５１】図４は、秘密s と部分行列S との関係を示す図である。

【００５２】本実施例において、ある秘密s に対する部分行列S とは、従来の対話型確認可能な秘密分散方式に用いられている部分行列に、以下に説明するように、更に条件が付けられた秘密部分のn ×n 行列S=〔s(i, j)〕(i, j=1, ･･･, n) である。

【００５３】すなわち、各行ベクトル（S_r(i)=〔s(i, 1), ･･･, s(i, n)〕, i=1, ･･･, n）が、ある秘密s_r(i)に対する秘密部分ベクトル（ベクトルの要素」が、ある秘密分散方式における加入者」の秘密部分である）になり、

各列ベクトル（S_c(j)=〔s(1, j), ･･･, s(n, j)〕, j=1, ･･･, n）が、ある秘密s_c(j)に対する秘密部分ベクトルになる。但し、前述の秘密のベクトル〔s_r(1), ･･･, s_r(n)〕及び〔s_c(1), ･･･, s_c(n)〕の両方は、元の秘密s に対する秘密部分ベクトルになる。

【００５４】秘密分散処理のときに、このように構成された部分行列の列ベクトルi （i=1, ･･･, n ）及び秘密s_r(i)を、加入者i の秘密部分情報として各加入者i に送信することによって、秘密復元処理のときに、部分情報を配った加入者が不正をしなければ、各加入者によって放送された部分情報が正しいかどうかを非常に高い確率で確認できる。

【００５５】また、本実施例では、元の秘密情報を持ち、秘密部分情報を配った加入者が不正をした場合でも、秘密復元処理のとき放送される部分情報が正しいかどうかを確認するために、秘密部分情報を配る上で各加入者i （i=1, ･･･, n ）に対応する秘密s_r(i)の認証子のような役割を果たす値を、一方向性ハッシュ関数（池野、小山：”現代暗号理論”、pp. 224-225、電子情報通信学会、1986、参照）によって生成する。

【００５６】次に、この一方向性ハッシュ関数について説明する。

【００５７】一方向性ハッシュ関数とは、データ圧縮型スクランブルを行う関数であり、入力値から出力値を求める演算は容易であるが、出力値から入力値を求める逆演算は困難である関数である。但し、本実施例では、従来の非対話型確認可能な秘密分散方式に用いられる一方向性関数に比べると、特殊な代数学的な性質を持たなくてもよく、計算処理の高速な一方向性ハッシュ関数を利用できる。

【００５８】一方向性ハッシュ関数の具体例として、R. Merkle によってDES （Data Encryption Standard）のようなブロック暗号を用いた一方向性ハッシュ関数が提案されている（"One Way Hash Functions and DES", Advances in Cryptology —Crypto'89, Lecture Notes in Computer Science, Vol. 435, Springer-Verlag, 1990参照）。

【００５９】図６は、一方向性ハッシュ関数の具体的な構成を説明する図である。

【００６０】同図において、(a) は、DES によるブロック暗号化を示しており、６１は、64ビットの入力及び56ビットの鍵から64ビットの出力が得られる暗号化回路（図８では、DES をE で表している）である。

【００６１】(b) は、このDES を部分処理として利用し、入力の長さが119 ビットで出力の長さが112 ビットとなる関数F の処理を示しており、６２は、関数演算回路である。この処理は、次のように定義される。

【００６２】まず、入力を二つの部分k, x に分ける（ただし、その一つの部分k 長さを55ビットとし、残りの部分x の長さを64ビットとする）。次に、その部分x をDES の入力とし、残りの部分k と'0' とを連続した値'0', k を56ビットの鍵として得られた出力と、x とのXOR を計算した結果を、関数F の出力における左側の64ビットとする。同様に、同じ部分x を入力とし、残りの部分k と'1' とを連続した値'1', k を鍵として得られた出力と、x とのXOR を計算した結果（64ビット）より48ビットを関数F の残りの部分（右側の48ビット）とする。これによって得られた二つの出力を連続した結果が112 ビットの関数F の出力となる。

【００６３】(c) は、与えられたメッセージを入力として、一方向性ハッシュ関数のハッシュ値を出力とする処理を示しており、６３は、ハッシュ関数処理部である。この処理は、次のように実行される。

【００６４】まず、与えられたメッセージの最初の119 ビットを上述の関数F の入力として、最初の112 ビットの出力を得る。次に、その出力を再び入力の112 ビット分とし、以後メッセージから残りの7 ビット分を連続して繰り返し関数F に入力する。最後に、全メッセージを入力したときに（メッセージの最後の7 ビットを入力するために足りないビットがあれば適当に'0' を加える）得られた112 ビットの出力を、メッセージに対するハッシュ値とする。

【００６５】前述のように計算されたハッシュ関数の一方向性（１つのハッシュ値が与えられたとき、同じハッシュ値が得られるような異なる入力のメッセージを求めることが極めて困難であること）は、用いられるDES のようなブロック暗号の安全性（入力が与えられたとき出力は鍵によるランダムな変数になり、出力が与えられたとき入力は鍵によるランダムな変数になる）によって与えられることが、R. Merkle より示されている（前述の論文参照）。

【００６６】更に、同じ論文では前述のハッシュ関数より効率のよい一方向性ハッシュ関数も提案されている。また、ブロック暗号を利用しない効率的な一方向性ハッシュ関数がR. Rivestによって提案されている（"The MD4 message digest algorithm", Advances in Cryptology —Crypto'90, Lecture Notes in Computer Science, Vol. 537, Springer-Verlag, 1991. NIST Federal Information Processing Standard for Secure Hash, American National Standard X9. 30-199x参照）。

【００６７】次に、本実施例におけるCut and Choose処理を説明する。

【００６８】従来の対話型確認可能な秘密分散方式で行われるCut and Chooseのように、秘密分散処理のとき全ての加入者が受け取った秘密部分情報が、確認可能に分散された秘密の部分情報となるかどうかを確認するために、元の秘密s に対する部分行列及びハッシュ値を分配すると同時にk 個のランダムに選ばれた秘密I1, ･･･, Ik に対する部分行列及びハッシュ値を分配し、全加入者によってランダムに決められたk/2 の秘密（Ii(1), ･･･, Ii

(k/2)) に関する全ての秘密情報を放送し、残りのk/2の秘密 (Ij(1), ･･･, Ij(k/2)) に対してIj(1)+s, ･･･, Ij(k/2)+s に関する全ての秘密情報を放送し、全ての放送された情報の中には誤りを持つ部分の数がt より多ければ、秘密分散処理が正しくないと判断される。

【００６９】以上によって、従来の確認可能な秘密分散方式の条件(c), (d) の代わりに、次の条件(c'), (d') を満たす秘密分散方式が実現でき、それを用いて安全に分散計算を実行する方式及び通信システムが実現できる。

【００７０】(c')不正な部分情報が正しい部分情報と混在したとき、t+1 個の正しい部分情報があっても元の秘密情報を復元できない場合が存在するが、不正をした加入者の識別は可能である。

【００７１】(d')全ての加入者がその秘密の部分情報を受け取ったとき、その部分情報がある秘密情報x を復元するための正しい情報でなければ、復元処理のときに不正をした加入者の識別が可能である。

【００７２】これらの条件は、従来方式のように不正な加入者があっても秘密を復元、すなわち誤り訂正は行えないが、不正な加入者は識別、すなわち誤り検出は可能にするものである。条件(c), (d) があれば不正な加入者がいても正しく分散演算が実行できる（正当性に対するフォールトトレランスを実現できる）が、条件(c'), (d') に変更しても不正な加入者は検出できるので警告などを行い再実行することによって正しい出力を得ることができ、正当性に対するフォールトトレランスを実現することができる。

【００７３】よって、本実施例では、分散された秘密を復元するとき不正な加入者があった場合には、復元できないこともありえるが、不正を行った加入者の識別は可能であるような確認可能な秘密分散方式を実現できる。

【００７４】そして、この秘密分散方式を利用して、署名者グループに参加する各加入者がランダムに選ばれた秘密の元をそのグループに参加する全加入者間で分散し、分散された秘密を入力として分散演算の実行から得られた分散出力の復元処理によって署名を生成する。

【００７５】従って、本実施例では、不正を行った加入者の識別が可能であり、計算量と通信量の両方が実用的なオーダーになる分散ディジタル署名方式が実現できる。

【００７６】図１は、本発明の１実施例である、分散した情報処理装置を有する情報処理システムを示す図である。

【００７７】同図において、１１は、システムの各加入者の利用する情報処理装置である。以下の説明では、各装置とそれを利用する各加入者とを同一視して、加入者と呼ぶことにする。１２は、全ての加入者に情報を公開することができる放送型通信路であり、１３は、各加入者毎に秘密の通信を行うことができる秘密通信路である。

【００７８】図２は、情報処理装置１１のブロック構成を示す図である。同図において、２１は、放送型通信路１２または秘密通信路１３により他の装置と通信を行なうための通信部である。２２は、演算処理部であり、記憶部２４のプログラムに従って、上述した一方向性ハッシュ関数などの各種演算や判定の処理を行なうとともに、装置各部を制御する。２３は、例えば疑似乱数発生器のような乱数発生部であり、ランダムな値を生成するために利用される。２４は、演算処理部２２が実行すべきプログラムや、処理の過程で生成される演算結果等の情報や、他の装置から受信した情報、各種パラメータなどを記憶するための記憶部である。

【００７９】以下、上述した装置構成により、分散ディジタル署名を実現する方法を具体的に説明する。

【００８０】〔実施例１〕この実施例では前述の C.P. Schnorr によって提案されたディジタル署名方式を用いた分散ディジタル署名方式を実現する具体的な構成について述べる。

【００８１】まず、 {1, ･･･, p-1 } の内にある秘密の元を確認可能に分散する具体的な構成を次に示す。

【００８２】ある与えられた秘密s に対する部分行列の具体的な説明を行う（図４参照）。 {1, ･･･, p-1 } の内にある秘密の元s に対する部分行列S=〔s(i, j)〕, i, j=1, ･･･, nとは、各行ベクトル (S_r(i)=〔s(i, 1), ･･･, s(i, n)〕, i=1, ･･･, n) の要素がs_r(i)を定数項とするt次の多項式fiのn 個の異なる値i1, ･･･, in に対する値fi(i1), ･･･, fi(in) になり、各列ベクトル (S_c(j)=〔s(1, j), ･･･, s(n, j)〕, j=1, ･･･, n) の要素がs_c(j)を定数項とするt 次の多項式gjのn 個の異なる値j1, ･･･, jnに対する値gj(j1), ･･･, gj(jn) になり、さらに前述の値のベクトル〔s_r(1), ･･･, s_r(n)〕 及び〔s_c(1), ･･･, s_c(n)〕 の両方は、元の秘密s を定数項とするt次の多項式f 及びg の値f(i1, ･･･, in)及びg(j1, ･･･, jn)になっているものである。

【００８３】秘密の元s を署名者グループに参加する全加入者間で秘密に分散保持されるように秘密部分を分配する秘密分散処理と、そのように分散された秘密または（不正があった場合に）不正をした加入者を明らかにする秘密復元処理の二つの処理に分けて説明する。

【００８４】以下に、h は効率的な（高速な計算方式のある）一方向性ハッシュ関数を表す。例えば高速なブロック暗号化関数によって構成されたハッシュ関数（前述の”現代暗号理論”参照）が用いられる。安全性パラメータk はある定数k'に対してk=nk' を満たす。この場合、秘密分散処理の確認が誤る確率は、Cut and Choose 処理によって（前記 T. Rabin, M. Ben-Or "Verifiable Secret Sharing and Multiparty Protocols with Honest Majority"参照 ）2^(-k'(t+1))になる。

【００８５】(1) 秘密分散処理（図５参照）：秘密の元s を持っている加入者d がs に対する秘密部分を分配す

る処理
（ラウンド１）加入者d は（図５では処理R1.dと表す）乱数発生手段を用いて、元の秘密s 及び｛1,･･･,p-1 ｝の内にあるランダムに選ばれた秘密の元I1,･･･,Ik に対する部分行列を生成し、秘密値s_r(1),･･･,s_r(n),I1_r(1),･･･,I1_r(n),･･･,Ik_r(1),･･･,Ik_r(n) に対するハッシュ関数の値s*を計算する（図７参照）。
【００８６】加入者d は各加入者i （i=1,･･･,n 、但し自分は除く）に秘密通信路を利用して、生成した各部分行列の各列ベクトルS_c(i),L1_c(i),･･･,Lk_c(i)及び秘密s_r(i),I1_r(i),･･･,Ik_r(i)（図５では情報B1.i）を秘密通信路を用いて送信し、全加入者間でハッシュ値s*（図５では情報B1.d）を、放送通信路を用いて放送する。
【００８７】（ラウンド２）各加入者i （i=1,･･･,n ）は（図５では処理R2.i）乱数発生手段を用いて、k'個のランダムに選択されたビットを放送する（図５ではB2.i）。それらのランダムに選ばれたk'個のビットはBi_1,..,Bi_k' 、n 人の全ての加入者に対する全ビットはB1,･･･,Bk' と呼ぶ。
【００８８】（ラウンド３）加入者d は（図５では処理R3.d）ラウンド２で放送された各ビットBj（j=1,･･･,k ）に対して、Bjが1 であればラウンド１で加入者d が生成した部分行列Ljを放送し、Bjが0 であればラウンド１で加入者d が生成した部分行列S とLjの要素毎の有限体上の加算結果（S+Ljと書く）を放送する。加入者d が放送した情報は、図５でB3.dと表される。
【００８９】（ラウンド４）各加入者i （i=1,･･･,n ）は（図５では処理R4.i）ラウンド１で秘密に受信した情報B1.iのなかに、各値j （j=1,･･･,k ）に対して列ベクトルLj_c(i) 及びIj_r(i) （ラウンド２で放送されたビットBjが1 の場合）あるいはLj_c(i)+S_c(i) 及びIj_r(i)+s_r(i)（Bjが0 の場合）がラウンド３で放送された部分行列に対応する列ベクトル及び秘密値と等しいかどうかを確認する。ある値j に対して、等しくなければ、加入者d の判定メッセージ（図５では情報B4.i）を放送する。
【００９０】（ラウンド５）加入者d は、ラウンド４で判定メッセージが放送された場合に、判定メッセージを放送した各加入者j に関してラウンド１で加入者d が秘密に送信した情報B1.j（図５では処理B5.d）を放送する（図５では処理R5.d）。
【００９１】（後処理）各加入者i （i=1,･･･,n ）は、ラウンド５で放送された情報が正しくなければ、あるいは、ラウンド４で放送された判定メッセージの個数が閾値tより多ければ、加入者d が不正を働いたと判断する（図５では処理Pi）。ラウンド１～５で各加入者i が受信した全ての情報をs_i と書く（図５参照）。
【００９２】(2) 秘密復元処理（図８参照）：秘密分散処理による各加入者が持っている情報s_i から、全ての加入者が元の秘密s を計算するための処理
（ラウンド１）各加入者i （i=1,･･･,n ）は（図８では処理R1.i）部分情報s_i 含まれている秘密値s_c(i)及びs_r(i)（図８では情報B1.i）を放送する。
【００９３】（ラウンド２）各加入者i （i=1,･･･,n ）は（図８では処理R2.i）ラウンド１で放送された値の内t+1 個の値s_c(i,1),･･･,s_c(i,t+1) 及びs_r(i,1),･･･,s_r(i,t+1) を選び、多項式補間処理の結果s(c)及びs(r)を求め、両方が等しくなり残りの放送された値がその値s(c)=s(r) に対応する同じ多項式の正しい値になるかどうかを確認する。全ての値が正しければ元の秘密sはs(c)=s(r) と等しいと判断し復元処理が終わる。同じ多項式に対応しない値が放送された場合に、部分情報s_i に含まれている列ベクトルS_c(i)を放送する（図８では情報B2.i）。
【００９４】（ラウンド３）各加入者i （i=1,･･･,n ）は（図８では処理R3.i）ラウンド２で放送された列ベクトルS_c(j)（j=1,･･･,n ）から部分行列S'を生成し、1,･･･,n ら異なるt+1 個の値を含む全ての集合t1,･･･,tm（全部でm=n!/((t+1)!(n-t-1)!) 個の集合がある）に対する列ベクトルの集合T1,･･･,Tm を選び、それぞれの行と列に対する多項式補間処理の結果s'_r(1),･･･,s'_r(n) 及びs'_c(1),･･･,s'_c(n) を求め、更にこれらの値の多項式補間処理の結果s'(r) 及びs'(c) を求め、両方が等しくなりそれらの列ベクトルの全ての要素がそれぞれの値s'_c(1),･･･,s'_c(n) 及びs'_r(1),..,s'_r(n)に対応する同じ多項式の正しい値になるかどうかを確認する。
【００９５】このように確認されたt+1 個の列ベクトルを含む集合T1,..,Tmの内の異なる正しい部分行列に対応する集合の数は最大t+1 個になり、それらの部分行列をS_1,･･･,S_T （ただし、T<:t+1 ）と呼ぶ。正しい部分行列が一個しかなければ（T=1）、その行列に対する秘密s が元の秘密と等しいと判断し、対応しない列ベクトルが不正をした加入者を表す。正しい部分行列が二つ以上があれば、各加入者i(i=1,･･･,n ）は持っている全ての部分情報s_i を放送する（図８では情報B3.i）。
【００９６】（後処理）各加入者i （i=1,･･･,n ）は（図８では処理Pi）ラウンド３で放送された情報s_j（j=1,･･･,n ）を用いて、ラウンド４で計算した正しい部分行列S_1,･･･,S_T （ただし、T<:t+1 ）に対して、秘密分散処理と同じ一方向性ハッシュ関数を計算し、全ての放送された情報を正しく、分散処理のラウンド１で放送された値s*に対応するかどうかを確認する。これらの確認に対して正しい（最大一つしかありえない）部分行列に対応する秘密s'は元の秘密s と等しいと判断する。それに対して前述の確認に対して正しい部分行列がなかったならば、秘密の分散を行った加入者d が不正したと判断する。
【００９７】以上により、署名者グループのある加入者

が持っている秘密の元を、そのグループの全加入者間で分散保持する処理が実現できる。次に、この秘密分散方式を用いて、分散ディジタル署名方式における署名者グループの秘密情報（秘密鍵に相当する、ただしそのグループに参加する全加入者間で分散されている）、及びその秘密情報に対応する公開情報（公開鍵に相当するそのグループによって生成された署名を確認するための公開情報）を生成する処理について述べる。

【００９８】(3) 鍵生成処理（図９参照）

（ラウンド１～５）各加入者i が {1,･･･,p-1 } の内にある秘密の元a(i)をランダムに選び、上述の秘密分散処理を利用して署名者グループに参加する全加入者間で分散する。さらに、各加入者i がg^(a(i)) mod q（ただし、q は前述のように選ばれた素数である）を計算し、放送通信路を利用して放送する。

【００９９】（ラウンド６）上述の秘密分散処理の後処理を実行し、正しく分散された秘密の元a(i)（i=1,･･･,n ）を入力として以下の分散加算によって得られた分散出力を秘密情報a とする。さらに、正しく分散された秘密の元a(i)に対する前の処理のラウンド１～５で放送された値 A(i)=g^(a(i)) mod qの乗算を計算し、得られた結果A=g^a mod q をそのグループによる署名を確認するための公開情報（公開鍵）とする。

【０１００】ラウンド６で用いた分散加算について図１０を参照にして説明する。

【０１０１】前述の秘密分散処理によって、 {1,･･･,p-1 } の内にある二つの秘密の元x とy 分散されたとき（各加入者i が秘密の元x とy に対応する秘密部分x_i 及びy_i 持つ）、通信を行わずに、 {1,･･･,p-1 } の内にある和x+y に対応する秘密部分(x+y)_i を次のように計算する。各加入者が持っている部分行列の列ベクトルX_c(i)及びY_c(i)かつ秘密値x_r(i)及びy_r(i)の要素毎の加算結果X_c(i)+Y_c(i) びx_r(i)+y_r(i) がx+y に対する部分行列の列ベクトル及び秘密値になることは、部分行列の定義（行と列の要素が多項式の値となる）から明らかである。

【０１０２】秘密復元処理において部分情報を確認するために用いられる一方向性ハッシュ関数の値x*及びy* は、両方を記憶し、加算結果x+y を復元処理のとき必要であれば、両方を用いることによって秘密x+y に対応する部分行列X+Y を確認する。同じように、分散された一つの秘密の元x と公開の元a との乗算を分散して計算するために、各加入者が持っている部分行列の各要素とaとの乗算を行った結果がx*a に対する部分行列の要素になる。

【０１０３】よって、上述の処理によって分散された二つの秘密の元x とy 及び公開の元aとb の線形結合a*x+b*y はそのグループに参加する加入者間で対話せずに分散して計算できる。この線形結合処理を図１１のように表現する。ただし、図は行う処理の名前と、その処理に対する各加入者毎の入出力を示している。各加入者i（i=1,･･･,n ）の入力は、分散された秘密の元x とy に対する部分情報x_i とy_i 及び公開の元a とb になり、出力は線形結合処理の結果z に対する部分情報z_i になる。

【０１０４】以上の処理によって得られた秘密鍵を利用して、与えられたメッセージm のディジタル署名がそのグループによって次のように分散して生成される。

【０１０５】(4) 署名生成処理（図１２参照）

（ラウンド１～５）各加入者i が {1,･･･,p-1 } の内にある秘密の元r(i)をランダムに選び、上述の秘密分散処理を利用して署名者グループに参加する全加入者間で分散する。さらに、各加入者i がg^(r(i)) mod q（ただし、q は前述のように選ばれた素数である）を計算し、放送通信路を利用して放送する。

【０１０６】（ラウンド６）上述の秘密分散処理の後処理を実行し、正しく分散された秘密の元r(i)（i=1,･･･,n ）を入力として先述の分散加算によって分散結果r を求める。さらに、正しく分散された秘密の元r(i)に対するラウンド１～５で放送された値 R(i)=g^(r(i)) mod q の乗算R=g^r mod q を計算する。

【０１０７】（ラウンド７）各加入者は与えられたメッセージm とラウンド６で計算された値R を連続した値R|m を入力として前述の所定の関数h の出力e=h(R|m)を計算し、署名者グループに参加する全加入者間で上述の分散線形結合処理を利用して、s=r+h(R|m)*aを分散して計算する。

【０１０８】（ラウンド８～１０）上述の秘密分散方式の秘密復元処理を利用して、分散秘密s を復元する。与えられたメッセージの署名は(R,s) とする。公開鍵a 及び前述のディジタル署名方式の署名確認処理を利用して、生成された署名を確認し、正しくないとき不正をした加入者が存在すると判断し、前述の秘密分散方式の秘密復元処理を実行することによって不正した加入者を識別する。

【０１０９】〔実施例２〕この実施例では前述のA. Fiat, A. Shamirによって提案されたディジタル署名方式を用いた分散ディジタル署名方式を実現する具体的な構成について述べる。まず、 {1,･･･,N-1 } の内にある秘密の元を確認可能に分散する具体的な構成を次に示す。

【０１１０】ある与えられた秘密s に対する部分行列の具体的な説明を行う（図４参照）。 {1,･･･,N-1 } の内にある秘密の元s に対する部分行列S=〔s(i,j)〕,i,j=1,･･･,nとは、各行ベクトル（S_r(i)=〔s(i,1),･･･,s(i,n)〕,i=1,･･･,n）の要素が次のように定義されるものである。

【０１１１】s(i,j)=s_r(i)*q_r(i,1)^(j)*q_r(i,2)^(j^2)*･･･*q_r(i,t)^(j^t) mod N (j=1,･･･,n)

ただし、q_r(i,1),･･･,q_r(i,t) は {1,･･･,N-1 } の内で以下の条件が満たされるように選ばれた元である。各

列ベクトル（S_c(j)=〔s(1, j), ･･･, s(n, j)〕, j=1, ･･･, n）の要素は次のように定義される。

【０１１２】s(j, i)=s_c(j)*q_c(j, 1)^(i)*q_c(j, 2)^(i^2)*･･･*q_c(j, t)^(i^t) mod N（i=1, ･･･, n）

ただし、q_c(j, 1), ･･･, q_c(j, t) は｛1, ･･･, N-1｝の内で以上の条件が満たされるように選ばれた元である。さらに前述の値のベクトル〔s_r(1), ･･･, s_r(n)〕及び〔s_c(1), ･･･, s_c(n)〕の両方は、｛1, ･･･, N-1｝の内にあるq_r(1), ･･･, q_r(t)、q_c(1), ･･･, q_c(t) に対して次の条件を満たす。

【０１１３】s_r(j)=s*q_r(1)^(j)*q_r(2)^(j^2)*･･･*q_r(t)^(j^t) mod N（j=1, ･･･, n）

s_c(i)=s*q_c(1)^(i)*q_c(2)^(i^2)*･･･*q_c(t)^(i^t) mod N（i=1, ･･･, n）

ただし、s は元の秘密を表す。

【０１１４】秘密の元s を署名者グループに参加する全加入者間で秘密に分散保持されるように秘密部分を分配する秘密分散処理と、そのように分散された秘密または（不正があった場合に）不正をした加入者を明かにする秘密復元処理の二つの処理が前述の実施例１の処理(1), (2) のように行われる。ただし、秘密復元処理(2) において行われる多項式補間処理の代わりに、各行ベクトル（S_r(i)=〔s(i, 1), ･･･, s(i, n)〕, i=1, ･･･, n）の内にあるt+1 個の要素（s(i, j0), ･･･, s(i, jt)）から値s_r(i) を求めるために、次の計算が行われる。

【０１１５】s_r(i)^(n!)=Prod_k(s(i, jk)^(Prod_l(l*n!/(l-k))))　mod N

ただし、Prod_k(f(k))はk=j0, ･･･, jt に対する値f(k)の乗算を表し、Prod_l(g(l))はl=j0, ･･･, jt （l ≠k）に対する値g(l)の乗算を表す。

【０１１６】同じように、各列ベクトル（S_c(i)=〔s(1, j), ･･･, s(n, j)〕, j=1, ･･･, n）の内にあるt+1 個の要素（s(i0, j), ･･･, s(it, j)）から値s_c(j)を求めるために、次の計算が行われる。

【０１１７】s_c(j)^(n!)=Prod_k(s(ik, j)^(Prod_l(l*n!/(l-k))))　mod N

これらの部分ベクトル〔s_r(1)^(n!), ･･･, s_r(n)^(n!)〕（あるいは〔s_c(1)^(n!), ･･･, s_c(n)^(n!)〕）の内にあるt+1 個の要素（s_r(i0)^(n!), ･･･, s_r(it)^(n!)、あるいは（s_c(j0)^(n!), ･･･, s_c(jt)^(n!)）から秘密s を求めるために次の計算が行われる（s_r(i)^(n!) の場合）。

【０１１８】s^(n!*n!)=Prod_k(s_r(k)^(n!*Prod_l(l*n!/(l-k))))　mod N

s=(s^(n!*n!))^u*(s^l)^v mod N

ただし、A. Fiat, A. Shamirによるディジタル署名方式に用いられるl は、u*n!*n!+v*l=1 になるようなu, v が存在するように選ばれる。分散された秘密s に対する値s^l mod N は、後に(6) において説明する署名処理において計算されている。

【０１１９】以上により、署名者グループのある加入者が持っている秘密の元をそのグループの全加入者間で分散保持する処理が実現できる。次に、この秘密分散方式を用いて分散ディジタル署名方式における署名者グループの秘密情報（秘密鍵に相当する、ただしそのグループに参加する全加入者間で分散されている）、及びその秘密情報に対応する公開情報（公開鍵に相当するそのグループによって生成された署名を確認するための公開情報）を生成する処理について述べる。

【０１２０】(5) 鍵生成処理（図１３参照）

（ラウンド１～５）各加入者i が｛1, ･･･, N-1｝の内にある秘密の元a(i)をランダムに選び、上述の秘密分散処理を利用して署名者グループに参加する全加入者間で分散する。さらに、各加入者i がa(i)^l mod N（ただし、l は前述のように選ばれた元である）を計算し、放送通信路を利用して放送する。

【０１２１】（ラウンド６）上述の秘密分散処理の後処理を実行し、正しく分散された秘密の元a(i)（i=1, ･･･, n）を入力として以下の分散乗算によって得られた分散出力を秘密情報a とする。さらに、正しく分散された秘密の元a(i)に対する前の処理のラウンド１～５で放送された値 A(i)=a(i)^l mod Nの乗算を計算し、得られた結果A=a^l mod N をそのグループによる署名を確認するための公開情報（公開鍵）とする。

【０１２２】ラウンド６で用いた分散乗算について図１４を参照にして説明する。

【０１２３】前述の秘密分散処理によって、｛1, ･･･, N-1｝の内にある二つの秘密の元x とy が分散されたとき（各加入者i が秘密の元x とy に対応する秘密部分x_i 及びy_i 持つ）、通信を行わずに、｛1, ･･･, N-1｝の内にある積x*y に対応する秘密部分(x*y)_i を次のように計算する。

【０１２４】各加入者が持っている部分行列の列ベクトルX_c(i)及びY_c(i)かつ秘密値x_r(i)及びy_r(i)の要素毎の乗算結果X_c(i)*Y_c(i) びx_r(i)*y_r(i) がx*y に対する部分行列の列ベクトル及び秘密値になることは、部分行列の定義から明らかである。秘密復元処理において部分情報を確認するために用いられる一方向性ハッシュ関数の値x*及びy*は、両方を記憶し、乗算結果x*y を復元処理のとき必要であれば、両方を用いることによって秘密x*y に対応する部分行列X*Y を確認する。

【０１２５】同じように、分散された一つの秘密の元x と公開の元a とのべき乗算を分散して計算するために、各加入者が持っている部分行列の各要素とa とのべき乗算を行った結果がx^a に対する部分行列の要素になる。よって、上述の処理によって分散された二つの秘密の元x とy 及び公開の元a b の結合x^a*y^b はそのグループに参加する加入者間で対話せずに分散して計算できる。これを図１１と同様に図１５のように表す。

【０１２６】以上の処理によって得られた秘密鍵を利用

して、与えられたメッセージm のディジタル署名がそのグループによって次のように分散して生成される。

【０１２７】(6) 署名生成処理（図１６参照）

（ラウンド１～５）各加入者i が {1,･･･,N-1 } の内にある秘密の元r(i)をランダムに選び、上述の秘密分散処理を利用して署名者グループに参加する全加入者間で分散する。さらに、各加入者i がr(i)^I mod N（ただし、I は前述のように選ばれた）を計算し、放送通信路を利用して放送する。

【０１２８】（ラウンド６）上述の秘密分散処理の後処理を実行し、正しく分散された秘密の元r(i)（i=1,･･･,n ）を入力として先述の分散乗算によって分散結果r を求める。さらに、正しく分散された秘密の元r(i)に対するラウンド１～５で放送された値 R(i)=r(i)^I mod Nの乗算R=r^I mod N を計算する。

【０１２９】（ラウンド７）各加入者は与えられたメッセージm とラウンド６で計算された値R を連続した値R|m を入力として前述の所定の関数h の出力e=h(R|m)を計算し、署名者グループに参加する全加入者間で上述の分散結合処理を利用して、s=r*a^(h(R|m))を分散して計算する。

【０１３０】（ラウンド８～１０）上述の秘密分散方式の秘密復元処理を利用して、分散秘密s を復元する。与えられたメッセージの署名は(R,s) とする。公開鍵a 及び前述のディジタル署名方式の署名確認処理を利用して、生成された署名を確認し、正しくないとき不正をした加入者が存在すると判断し、前述の秘密分散方式の秘密復元処理を実行することによって不正した加入者を識別する。

【０１３１】〔実施例３〕この実施例では前述のT. ElGamal (American National Standard, Digital Signature Algorithm ）によって提案されたディジタル署名方式を用いた分散ディジタル署名方式を実現する具体的な構成について述べる。

【０１３２】署名者グループのある加入者が持っている秘密の元をそのグループの全加入者間で分散保持する具体的な方式として、実施例１に示した方式を利用する。その秘密分散方式を用いて分散ディジタル署名方式における署名者グループの秘密情報（秘密鍵に相当する、ただしそのグループに参加する全加入者間で分散されている）、及びその秘密情報に対応する公開情報（公開鍵に相当するそのグループによって生成された署名を確認するための公開情報）を生成する処理について次に述べる。

【０１３３】(7) 鍵生成処理（図９参照）

（ラウンド１～５）各加入者i が {1,･･･,p-1 } の内にある秘密の元a(i)をランダムに選び、上述の秘密分散処理を利用して署名者グループに参加する全加入者間で分散する。さらに、各加入者i がg^(a(i)) mod q（ただし、q は前述のように選ばれた素数である）を計算し、放送通信路を利用して放送する。

【０１３４】（ラウンド６）上述の秘密分散処理の後処理を実行し、正しく分散された秘密の元a(i)（i=1,･･･,n ）を入力として以下の分散加算によって得られた分散出力を秘密情報a とする。さらに、正しく分散された秘密の元a(i)に対するこの処理のラウンド１～５で放送された値 A(i)=g^(a(i)) mod qの乗算を計算し、得られた結果A=g^a mod q をそのグループによる署名を確認するための公開情報（公開鍵）とする。

【０１３５】前述の秘密分散処理によって、 {1,･･･,p-1 } の内にある二つの秘密の元x とy 分散されたとき（各加入者i が秘密の元x とy に対応する秘密部分x_i 及びy_i 持つ）、通信を行わずに、 {1,･･･,p-1 } の内にある和x+y に対応する秘密部分(x+y)_i を実施例１に示したように計算する。

【０１３６】前述の秘密分散処理によって、 {1,･･･,p-1 } の内にある二つの秘密の元x とy 分散されたとき（各加入者i が秘密の元x とy に対応する秘密部分x_i 及びy_i 持つ）、通信を行わずに、 {1,･･･,p-1 } の内にある積x*y に対応する秘密部分(x*y)_i を次のように計算する。

【０１３７】まず、ある加入者d が、 {1,･･･,p-1 } の内にある秘密s を確認可能な分散を行いながら、その秘密s と以前分散されたx の分散された積s*x を安全に計算する。その処理を確認可能な秘密分散及び乗算処理と呼ぶ。

【０１３８】(8) 確認可能な秘密分散及び乗算処理（図１７、１８参照）

以下において、実施例１と同じように、h は効率的な一方向性ハッシュ関数を表す。安全性パラメータk はある定数k'に対してk=nk' を満たす。

【０１３９】（ラウンド１）加入者d は乱数発生手段を用いて（図１７では処理R1.dと表す）、秘密の元s 及び {1,･･･,p-1 } の内にあるランダムに選ばれた秘密の元 I1,･･･,Ik に対する部分行列を生成し、秘密値s_r(1),･･･,s_r(n),I1_r(1),･･･,I1_r(n),･･･,Ik_r(1),･･･,Ik_r(n) に対するハッシュ関数の値s*を計算する（図７参照）。加入者d は各加入者i （i=1,･･･,n 、ただし自分は除く）に秘密通信路を利用して、生成した各部分行列の各列ベクトルS_c(i),L1_c(i),･･･,Lk_c(i)及び秘密s_r(i),I1_r(i),･･･,Ik_r(i)を秘密通信路を用いて送信し、全加入者間にハッシュ値s*を(図１７でB1.d) 、放送通信路を用いて放送する。

【０１４０】（ラウンド２）各加入者i （i=1,･･･,n ）（図１７では処理R2.i）は、ラウンド１で受け取った秘密s_r(i),I1_r(i),･･･,Ik_r(i)及び以前分散されたx に対する秘密x_r(i)を入力として、t_r(i)=s_r(i)*x_r(i), m1_r(i)=I1_r(i)*x_r(i),･･･, mk_r(i)*x_r(i) mod Nの計算を行い、乱数発生手段を用いて、得られた結果に対する部分行列 (T(i),M1(i),･･･,Mk(i) と呼び、

以前分散された秘密x の部分x_r(i)に対する部分行列をX(i)と呼ぶ）を生成し、秘密値t(i)_r(1),･･･,t(i)_r(n),m1(i)_r(1),･･･,m1(i)_r(n),･･･,mk(i)_r(1),･･･,mk(i)_r(n) に対するハッシュ関数の値s*を計算する（図７参照）。各加入者i は各加入者j （j=1,･･･,n 、ただし自分は除く）に秘密通信路を利用して、生成した各部分行列の各列ベクトルT(i)_c(j),M1(i)_c(j),･･･,Mk(i)_c(j) 及び秘密t(i)_r(j),m1(i)_r(j),･･･,mk(i)_r(j) を秘密通信路を用いて送信し、全加入者間でハッシュ値s*を、放送通信路を用いて放送する。加入者i が放送した情報は、図１７でB2.iと表される。

【０１４１】（ラウンド３）各加入者j （j=1,･･･,n ）（図１７では処理R3.i）は、乱数発生手段を用いて、k'個のランダムに選択されたビットを放送する。それらのランダムに選ばれたk'個のビットをBj_1,..,Bj_k' 、n 人の全ての加入者に対する全ビットをB1,･･･,Bk と呼ぶ。加入者i が放送した情報は、図１７でB3.iと現される。

【０１４２】（ラウンド４）加入者d （図１７では処理R4.d）は、ラウンド３で放送された各ビットBj（j=1,･･･,k ）に対して、Bjが1 であればラウンド１で加入者d が生成した部分行列Ljを放送し、Bjが0 であればラウンド１で加入者d が生成した部分行列S とLjの要素毎の有限体上の加算結果（S+Ljと書く）を放送する（ 図１７でB4.d）。

【０１４３】（ラウンド５）各加入者i （i=1,･･･,n ）（図１７では処理R5.i）は、ラウンド１で秘密に受信した情報B1.iの中のj （j=1,･･･,k ）に対する各列ベクトルLj_c(i) 及びlj_r(i) （ラウンド２で放送されたビットBjが1 の場合）あるいは Lj_c(i)+S_c(i) 及びlj_r(i)+s_r(i)（Bjが0 の場合）がラウンド４で放送された部分行列に対応する列ベクトル及び秘密値と等しいかどうかを確認する。ある値j 対して等しくなければ加入者d の判定メッセージを放送する。加入者i が放送した情報は、図１７、１８でB5.iと表される。

【０１４４】（ラウンド６）各加入者i （図１８では処理R6.i）は、ラウンド３で放送された各ビットBj（j=1,･･･,k ）に対して、Bjが1 であればラウンド２で加入者i が生成した部分行列 Mj(i)を放送し、Bjが0 であればラウンド１で加入者d が生成した部分行列 T(i) とMj(i) の要素毎の有限体上（mod p ）の加算結果（T(i)+Mj(i)と書く）を放送する。更に、加入者d は、ラウンド５で判定した加入者の列ベクトルを放送する。加入者i が放送した情報は、図１８でB6.iと表される。

【０１４５】（ラウンド７）各j に対して（j=1,･･･,k ）、各加入者i （図１８では処理R7.i）は、ラウンド５～６で放送された情報を確認する。しきい値t より多くの行列が正しく、かつ、ラウンド３で放送された各ビットBj（j=1,･･･,k ）に対して、Bjが1 であれば次の部分行列を放送する。

【０１４６】(lj_r(i))^(-1)*Mj(i)-X(i)

また、Bjが0 であれば次の部分行列を放送する。

【０１４７】

(s_r(i)+lj_r(i))^(-1)*(T(i)+Mj(i))-X(i)

ただし、部分行列の線形結合は前述のように（要素毎に）行われる。加入者iが放送した情報は、図１８でB7.iと現される。

【０１４８】（ラウンド８）各j,o に対して（j,o=1,･･･,k ）、各加入者i （i=1,･･･,n ）（図１８では処理R8.i）は、ラウンド４で放送された情報を利用して、Boが1 であればlj_r(o) を復元した後に次の列ベクトルを計算し、列ベクトル結果が正しくなること、及び値0 に対応することを確認する。

【０１４９】

(lj_r(o))^(-1)*Mj(o)_r(i)-X(o)_r(i)

Boが0 であれば(s_r(o)+lj_r(o))を復元した後に次の列ベクトルを計算し、列ベクトル結果が正しくなること、及び値0 に対応することを確認する。

【０１５０】(s_r(o)+lj_r(o))^(-1)*(T(o)_r(i)+Mj(o)_r(i))-X(o)_r(i)

ある値o に対して確認できなければ加入者o の判定メッセージを放送する。加入者i が放送した情報は、図１８でB8.iと現される。

【０１５１】（ラウンド９）各加入者o （図１８では処理R9.o）は、ラウンド８で自分に対して判定メッセージが放送された場合に、その列ベクトルを放送する。加入者iが放送した情報は、図１８でB9.iと現される。

【０１５２】（ラウンド１０）各加入者i （図１８では処理R10.i ）は、全ての放送された情報を確認し、不正を行った他の加入者o に対して判定メッセージを放送する。加入者i が放送した情報は、図１８でB10.i と現される。

【０１５３】（後処理）全加入者によって（各加入者iの処理が図１８では処理Piで表される）正しく分散された部分行列S(i)*X(i) の分散線形結合を行うことによって正しい部分行列S*X を分散して計算する。不正によってその行列を生成できない場合には、不正を行った加入者を識別し、その結果を放送する。

【０１５４】以上の処理を行うことによって、ある加入者 i がランダムに選んだ {1,･･･,p-1 の内にある秘密の元s は、確認可能な形で分散が行われながら、同時にその秘密s と以前に確認可能に分散された秘密x との積s*x が計算される。次に、上述の処理を利用して構成される二つの分散されている秘密x とy の積を実現する計算についてのべる。これを図１９に示す。

【０１５５】（ラウンド１～１０）各加入者 i がランダムに {1,･･･,p-1 } の内にある秘密の元r(i)を選択し、上述の処理によって確認可能な分散を行いながらr(i)*yを同時に計算する。

【０１５６】（ラウンド１１）正しく確認可能な分散が

行われている全ての加入者jl(l=1,･･･,m) に対する秘密の加算r = r(j1)+r(j2)+･･･+r(jm) と r*y = r(j1)*y+r(j2)*y+･･･+r(jm)*yが前述の分散加算によって計算される。

【０１５７】（ラウンド１２）秘密の加算u=x-r が分散加算される。

【０１５８】（ラウンド１３～１５）全ての加入者によって、秘密u が復元される。

【０１５９】（ラウンド１６）秘密の線形結合z=r*y-u*y=x*y が分散計算される。

【０１６０】上述の処理によって分散された二つの秘密の元x とy の積x*y はそのグループに参加する加入者間で分散して計算できる。以上の処理によって得られた秘密鍵を利用して、与えられたメッセージm のディジタル署名がそのグループによって次のように分散して生成される。

【０１６１】(8) 署名生成処理（図２０参照）

（ラウンド１～５）各加入者i が {1,･･･,p-1 } の内にある秘密の元r(i)をランダムに選び、上述の秘密分散処理を利用して署名者グループに参加する全加入者間で分散する。さらに、各加入者i がg^(r(i)) mod q（ただし、q は前述のように選ばれた素数である）を計算し、放送通信路を利用して放送する。

【０１６２】（ラウンド６）上述の秘密分散処理の後処理を実行し、正しく分散された秘密の元r(i)（i=1,･･･,n ）を入力として先述の分散加算によって分散結果r を求める。さらに、正しく分散された秘密の元r(i)に対するラウンド１～５で放送された値 R(i)=g^(r(i)) mod q の乗算R=g^r mod q を計算する。

【０１６３】（ラウンド７）各加入者は与えられたメッセージm を入力として前述の所定の関数h の出力e=h(m)を計算し、署名者グループに参加する全加入者間で上述の分散線形結合処理を利用して、b=(e+R*a) を分散して計算する。

【０１６４】（ラウンド８～１７）署名者グループに参加する全加入者間で上述の分散乗算処理を利用して、s=b*r^(-1)を分散して計算する。ただし、r^(-1)の分散逆元処理を避けるためには、乱数生成処理（ラウンド１～５参照）においてg^(r(i)) mod qの代わりにg^((r(i))^(-1)) mod q を計算すればよい。

【０１６５】（ラウンド１８～２０）上述の秘密分散方式の秘密復元処理を利用して、分散秘密s を復元する。与えられたメッセージの署名は(R,s) とする。公開鍵a及び前述のディジタル署名方式の署名確認処理を利用して、生成された署名を確認し、正しくないとき不正をした加入者が存在すると判断し、前述の秘密分散方式の秘密復元処理を実行することによって不正した加入者を識別する。

【０１６６】〔他の実施例〕本発明は、実施例１～３に示した部分行列の次元数等に限定されるものではなく、さらに多次元の部分配列でもよく、用いる関数は一方向性ハッシュ関数でなくても、一方向性が保証されれば他の関数でもよい。また、Cut and Choose技術も実施例１に具体的に示した手順でなくても、全ての秘密を漏らすことなくその正当性を確認できる手法であればよい。

【０１６７】以上説明したように、本実施例では、通信システムによって接続されている複数の計算装置（署名者）からなる署名者グループによって、そのグループの中のあるしきい値t より多くの署名者が正しければ署名を生成でき、不正があった場合には不正をした署名者を識別できるという意味の分散ディジタル署名方式を前述の通り構成できる。この方式は従来技術による不正があっても正しい署名を生成できるという分散ディジタル署名方式に比べて、通信量と計算量がより効率的になっている。

【０１６８】また、従来技術による不正があった場合に不正者が識別できず署名も生成できないという分散ディジタル署名方式に対しては、不正を行った署名者を識別できるという点において、本発明による方式の方が安全であると言える。

【０１６９】具体的に、ある署名者グループに参加する各署名者に必要な計算量は、一番計算量がかかる処理がべき乗演算となるので、従来の後者のディジタル署名方式においてある署名者に必要な計算量とほぼ同じと考えられる。しかし、各署名者に必要な通信量は、l*n^2*k のオーダー（n は加入者の数、k は安全パラメータ、l は用いられる整数の長さ）になり、従来の前者のディジタル署名方式に比べて実用的であると考えられる。

【０１７０】ただし、秘密復元処理（前述の実施例１の処理(2) のラウンド３参照）で行うm=n!/((t+1)!(n-t-1)!) の異なるt+1 列ベクトルを含む集合の数によって、n は十分小さく（n <20）なければ、不正を行った加入者の識別は実用上で困難であると考えられる。よって、本実施例による秘密分散方式は加入者の数が小さい場合に対して効果的であると言える。しかし、従来の確認可能な秘密分散方式は加入者の数が小さい場合に対しても非実現的な通信量と計算量であった。

【０１７１】

【発明の効果】以上説明したように、本発明による署名生成方法を用いることによって通信量及び計算量が削減できる。また、これにより、通信量が少ないことによって通信システムのトラフィックや通信料金等が改善され、計算量が少ないことによって処理が高速化されるという効果を生じる。

【図面の簡単な説明】

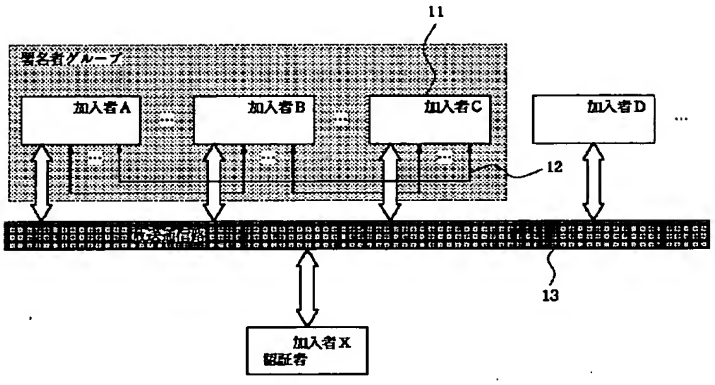
【図１】本発明の１実施例の通信システムの機能構成を示すブロック図である。

【図２】情報処理装置のブロック構成を示す図である。

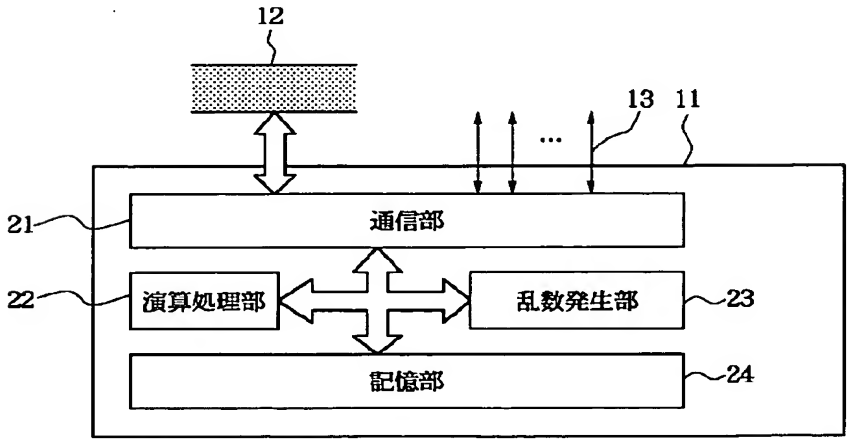
【図３】確認可能な秘密分散方式の手順を示す図である。

【図４】秘密部分行列の説明図である。
【図５】秘密分散処理手順の説明図である。
【図６】一方向性ハッシュ関数の具体例を示す図である。
【図７】一方向性ハッシュ関数の説明図である。
【図８】秘密復元処理手順の説明図である。
【図９】秘密鍵及び公開鍵生成処理手順の説明図である。
【図１０】分散秘密の加算処理の説明図である。
【図１１】分散秘密の線形結合処理の入出力関係の説明図である。
【図１２】分散署名生成処理手順の説明図である。
【図１３】秘密鍵及び公開鍵生成処理手順の説明図である。
【図１４】分散秘密の乗算処理の説明図である。
【図１５】分散秘密の結合処理の入出力関係の説明図である。
【図１６】分散署名生成処理手順の説明図である。
【図１７】確認可能な秘密及び分散乗算処理手順の説明図である。
【図１８】確認可能な秘密及び分散乗算処理手順の説明図である。
【図１９】分散秘密の乗算処理手順の説明図である。
【図２０】分散署名生成処理手順の説明図である。
【符号の説明】
１１　情報処理装置
１２　放送通信路
１３　秘密通信路
２１　通信部
２２　演算処理部
２３　乱数発生部
２４　記憶部
６１　暗号化回路
６２　関数演算回路
６３　ハッシュ演算処理部

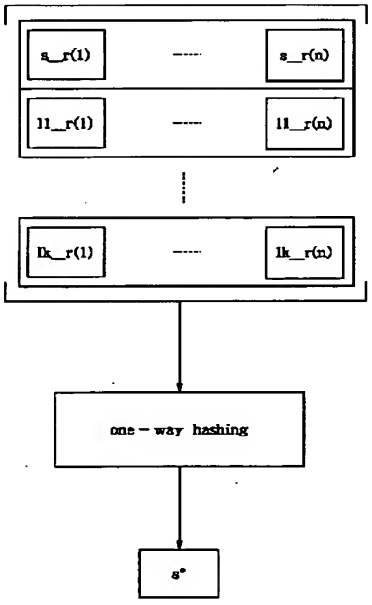
【図１】



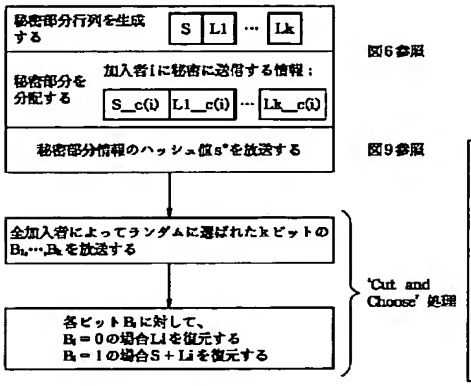
【図２】



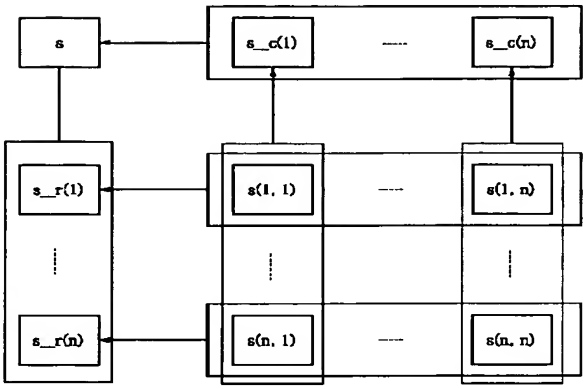
【図７】



【図３】



【図４】



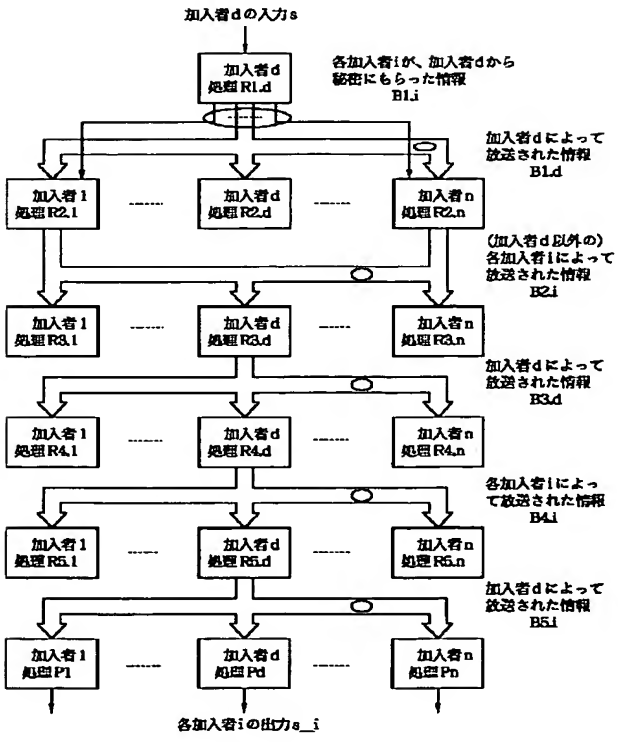
$$S=\begin{bmatrix} s(1,1) & \cdots & s(1,n) \\ \vdots & & \vdots \\ s(n,1) & \cdots & s(n,n) \end{bmatrix}$$

$$S\_c(1)=\begin{bmatrix} s(1,1) \\ \vdots \\ s(n,1) \end{bmatrix} \cdots S\_c(n)=\begin{bmatrix} s(1,n) \\ \vdots \\ s(n,n) \end{bmatrix}$$

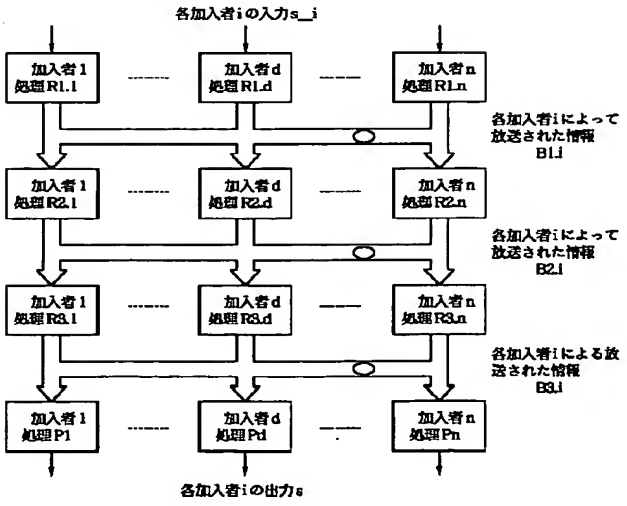
$$S\_r(1)=[s(1,1) \cdots s(1,n)]$$

$$S\_r(n)=[s(n,1) \cdots s(n,n)]$$

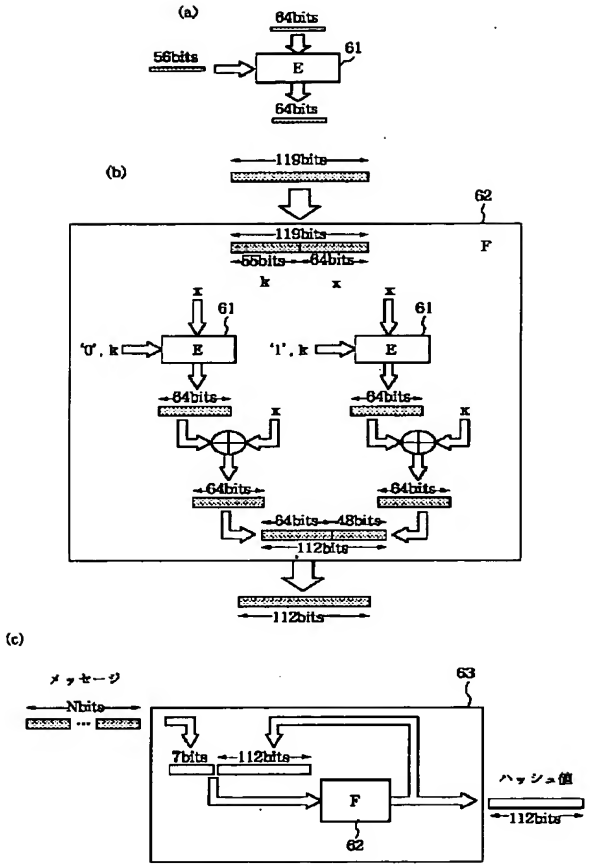
【図５】



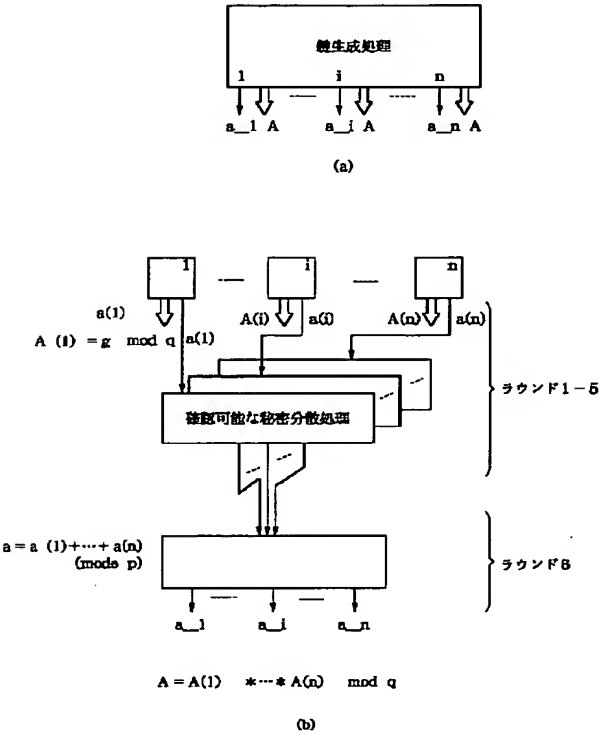
【図８】



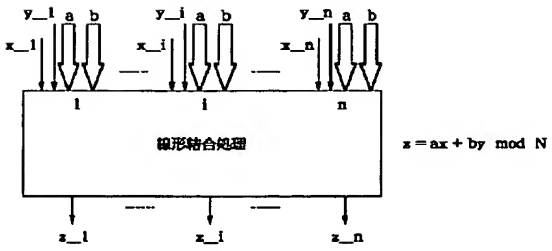
【図６】


(a)
64bits
56bits
E
61
64bits
(b)
119bits
62
F
55bits
8bits
k
x
'0', k
'1', k
64bits
48bits
112bits
(c)
メッセージ
Nbits
63
7bits
ハッシュ値
62

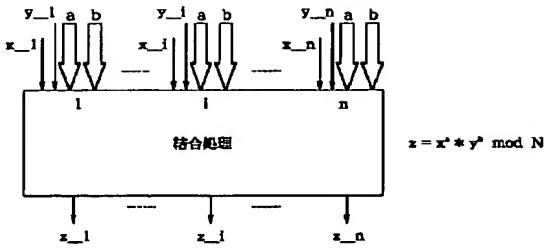
【図９】


鍵生成処理
a_1 A
a_i A
a_n A
(a)
a(1)
A(i)
a(i)
A(n)
a(n)
A (1) = g mod q a(1)
確認可能な秘密分散処理
ラウンド1－5
a = a (1)+…+ a(n)
(mode p)
ラウンド6
A = A(1) *…* A(n) mod q
(b)

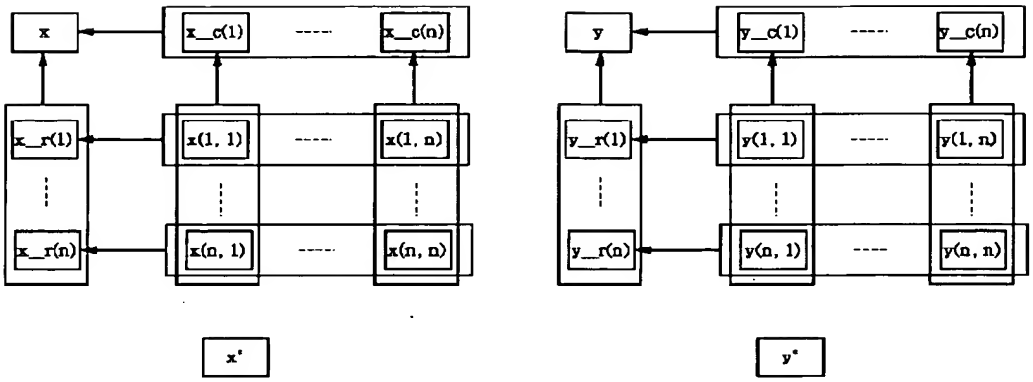
【図１１】


線形結合処理
z = ax + by mod N

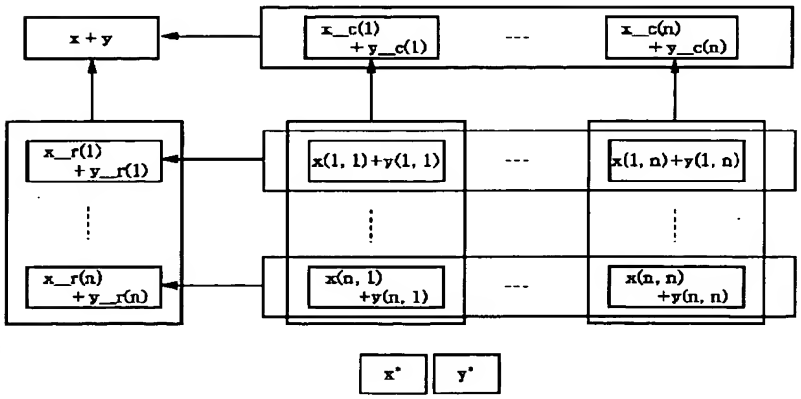
【図１５】


結合処理
z = x^a * y^b mod N

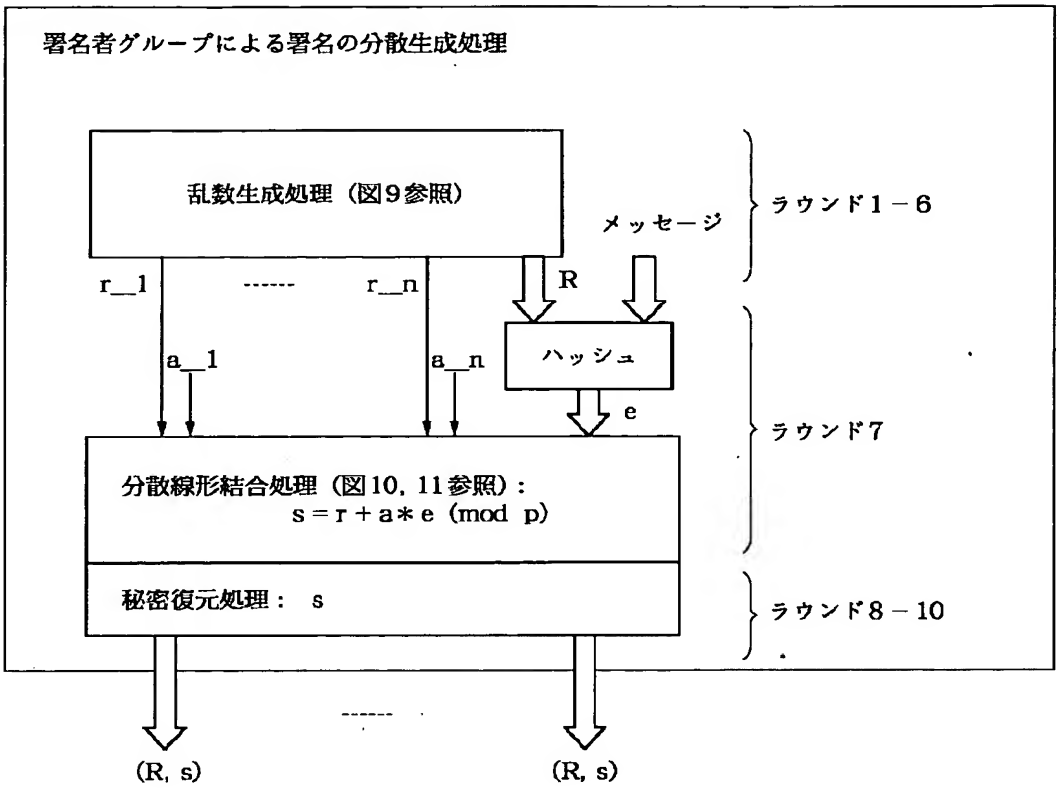
【図１０】


x
x_c(1)
x_c(n)
x_r(1)
x_r(n)
x(1, 1)
x(1, n)
x(n, 1)
x(n, n)
x*
y
y_c(1)
y_c(n)
y_r(1)
y_r(n)
y(1, 1)
y(1, n)
y(n, 1)
y(n, n)
y*

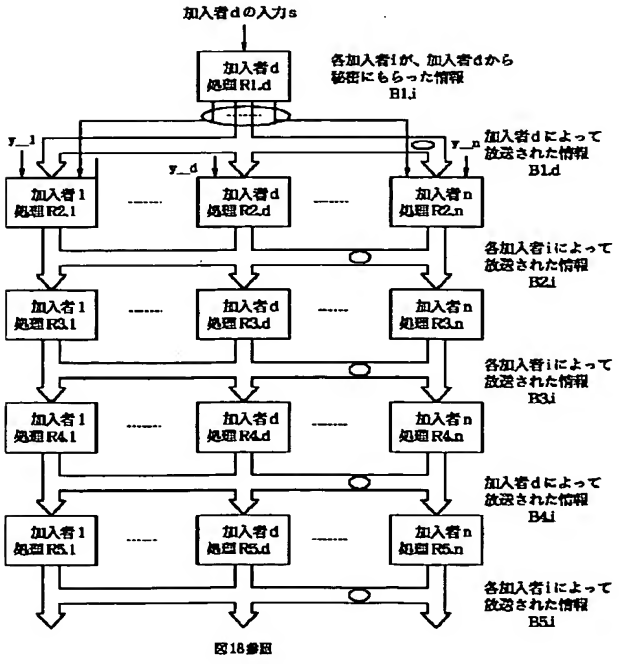

加算


x + y
x_c(1) + y_c(1)
x_c(n) + y_c(n)
x_r(1) + y_r(1)
x_r(n) + y_r(n)
x(1, 1)+y(1, 1)
x(1, n)+y(1, n)
x(n, 1) +y(n, 1)
x(n, n) +y(n, n)
x*
y*

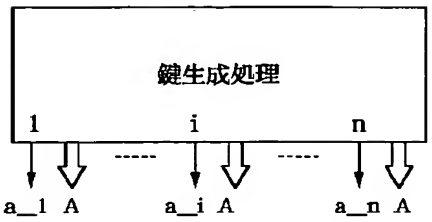
【図１２】


署名者グループによる署名の分散生成処理
乱数生成処理（図9参照）
メッセージ
ラウンド1－6
r__1
r__n
R
a__1
a__n
ハッシュ
e
ラウンド7
分散線形結合処理（図10，11参照）：
s＝r＋a＊e（mod p）
秘密復元処理： s
ラウンド8－10
(R, s)
(R, s)

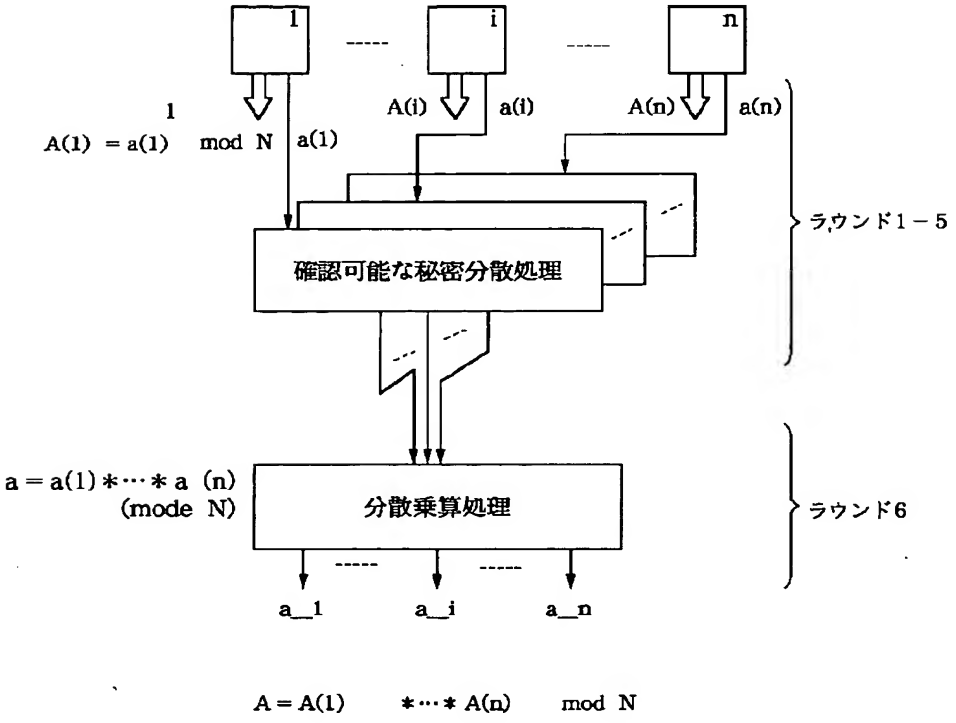
【図１７】


加入者dの入力s
加入者d 処理R1.d
各加入者iが、加入者dから秘密にもらった情報 B1.i
加入者dによって放送された情報 B1.d
加入者1 処理R2.1
加入者d 処理R2.d
加入者n 処理R2.n
各加入者iによって放送された情報 B2.i
加入者1 処理R3.1
加入者d 処理R3.d
加入者n 処理R3.n
各加入者iによって放送された情報 B3.i
加入者1 処理R4.1
加入者d 処理R4.d
加入者n 処理R4.n
加入者dによって放送された情報 B4.i
加入者1 処理R5.1
加入者d 処理R5.d
加入者n 処理R5.n
各加入者iによって放送された情報 B5.i
図18参照

【図１３】



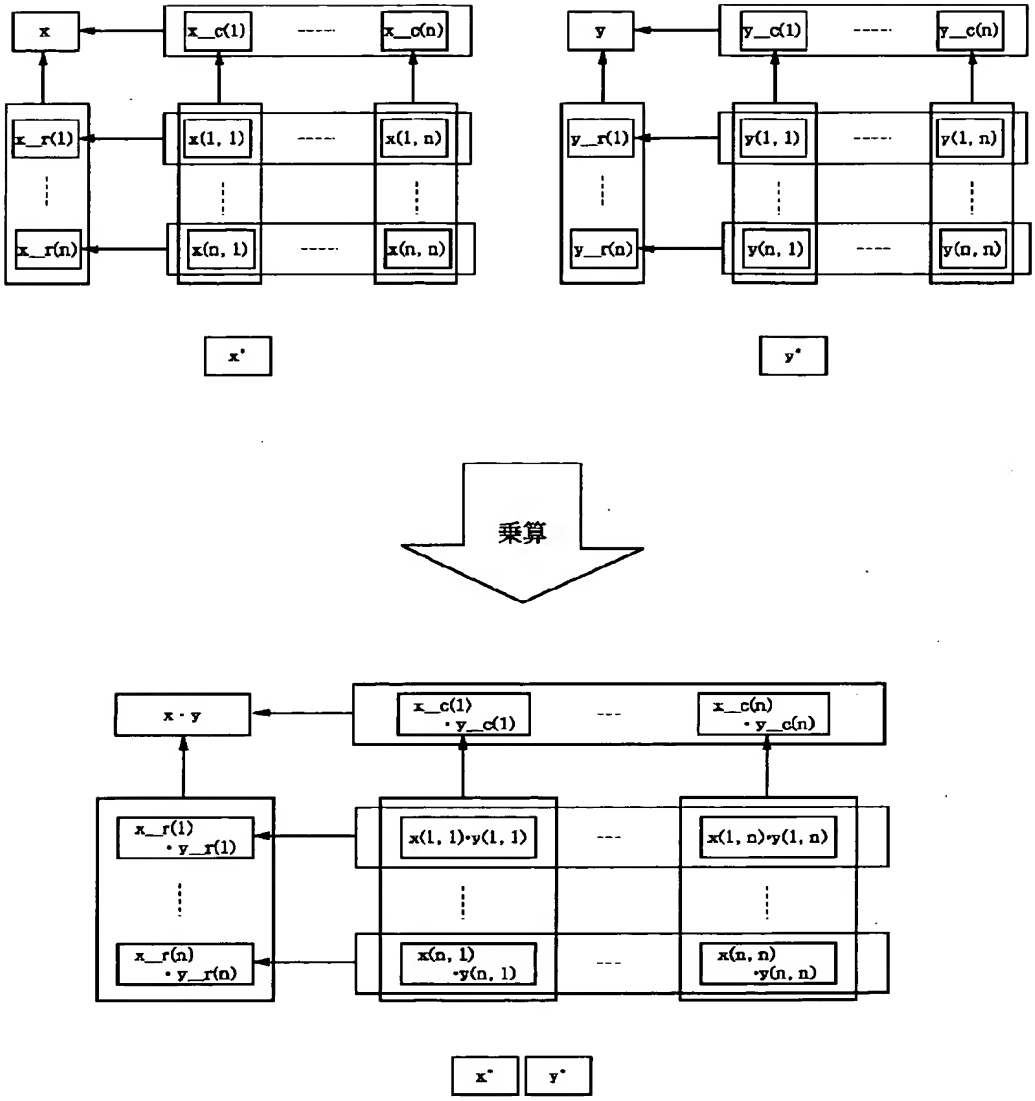
(a)



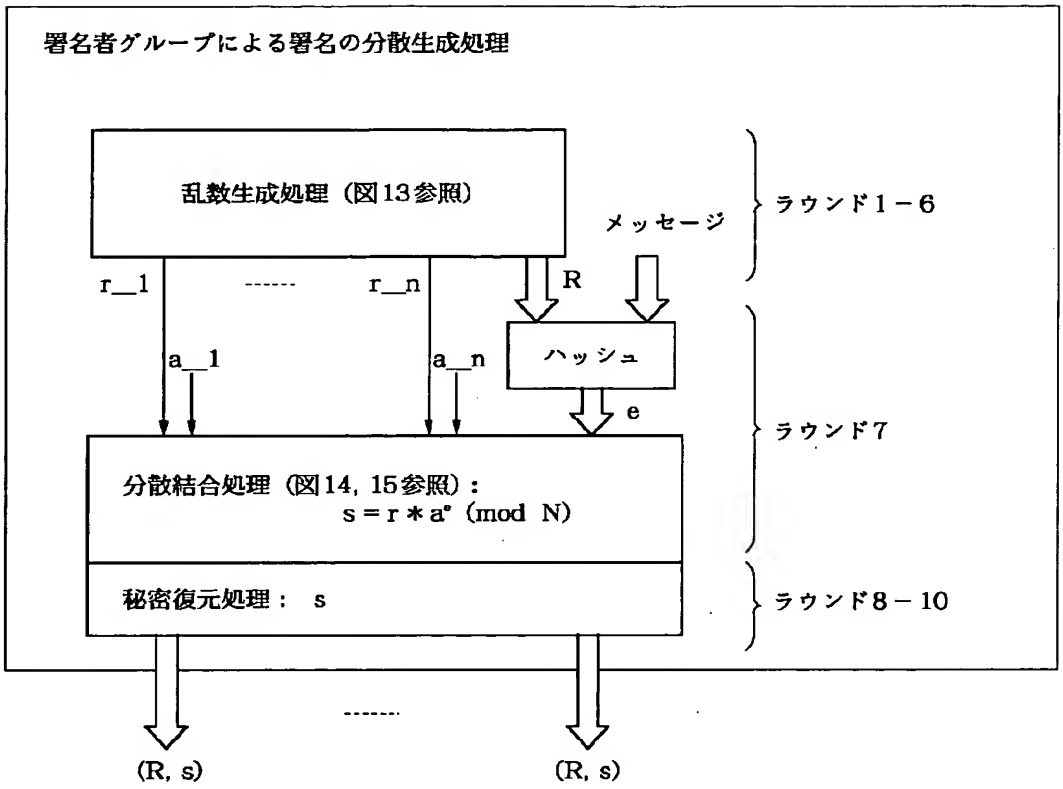
A = A(1) ＊…＊ A(n) mod N

(b)

【図１４】


x
x_c(1)
x_c(n)
x_r(1)
x_r(n)
x(1, 1)
x(1, n)
x(n, 1)
x(n, n)
x*
y
y_c(1)
y_c(n)
y_r(1)
y_r(n)
y(1, 1)
y(1, n)
y(n, 1)
y(n, n)
y*
乗算
x・y
x_c(1)・y_c(1)
x_c(n)・y_c(n)
x_r(1)・y_r(1)
x_r(n)・y_r(n)
x(1, 1)・y(1, 1)
x(1, n)・y(1, n)
x(n, 1)・y(n, 1)
x(n, n)・y(n, n)
x*
y*

【図１６】


署名者グループによる署名の分散生成処理
乱数生成処理（図13参照）
メッセージ
ラウンド1－6
r__1
r__n
R
a__1
a__n
ハッシュ
e
ラウンド7
分散結合処理（図14, 15参照）：
s＝r＊a^e（mod N）
秘密復元処理： s
ラウンド8－10
(R, s)
(R, s)

【図１８】

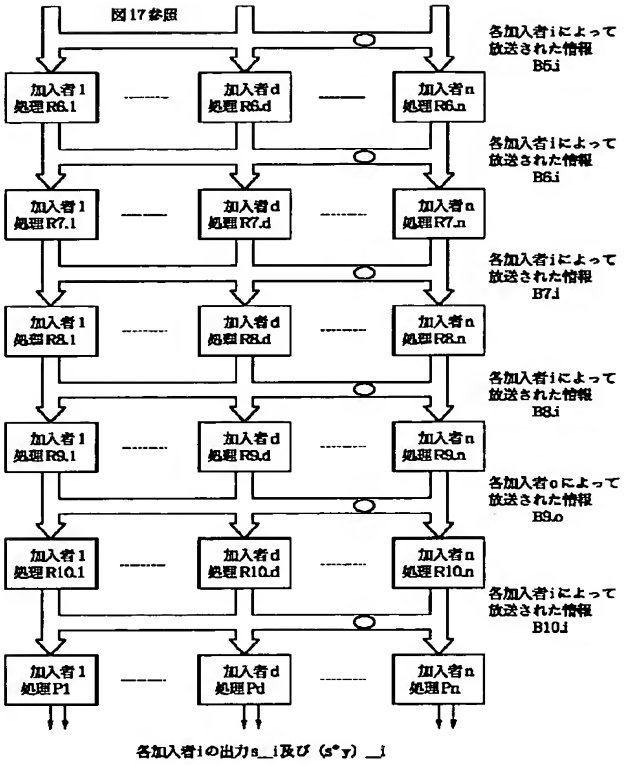
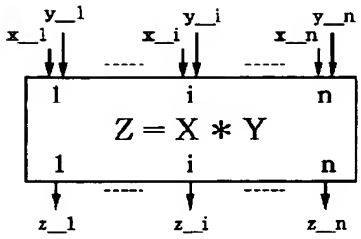
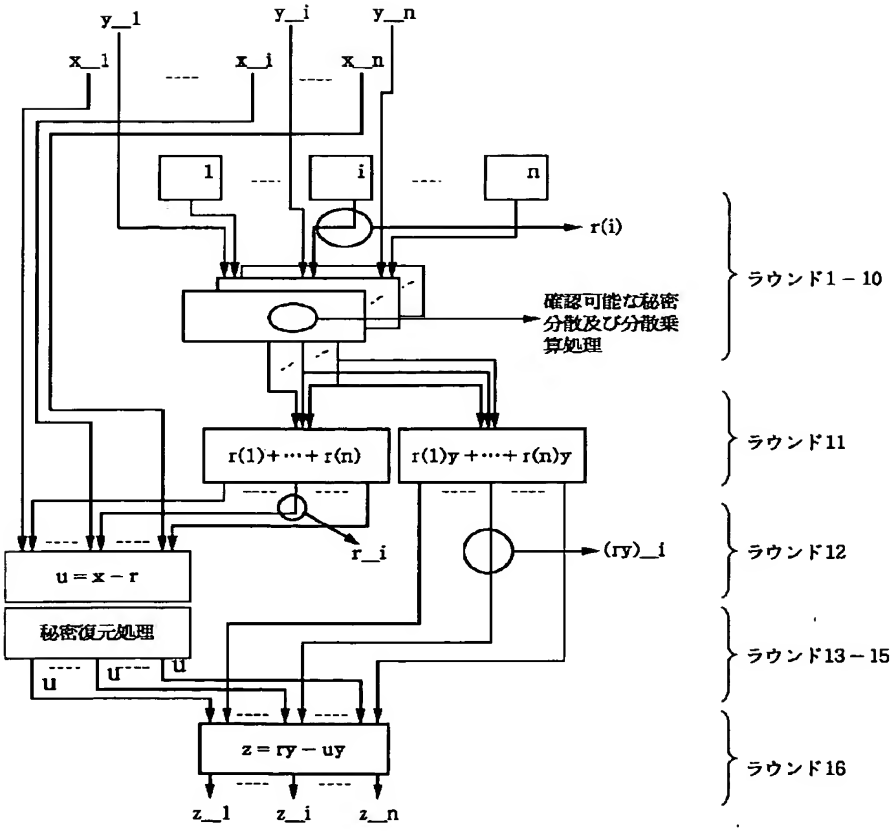


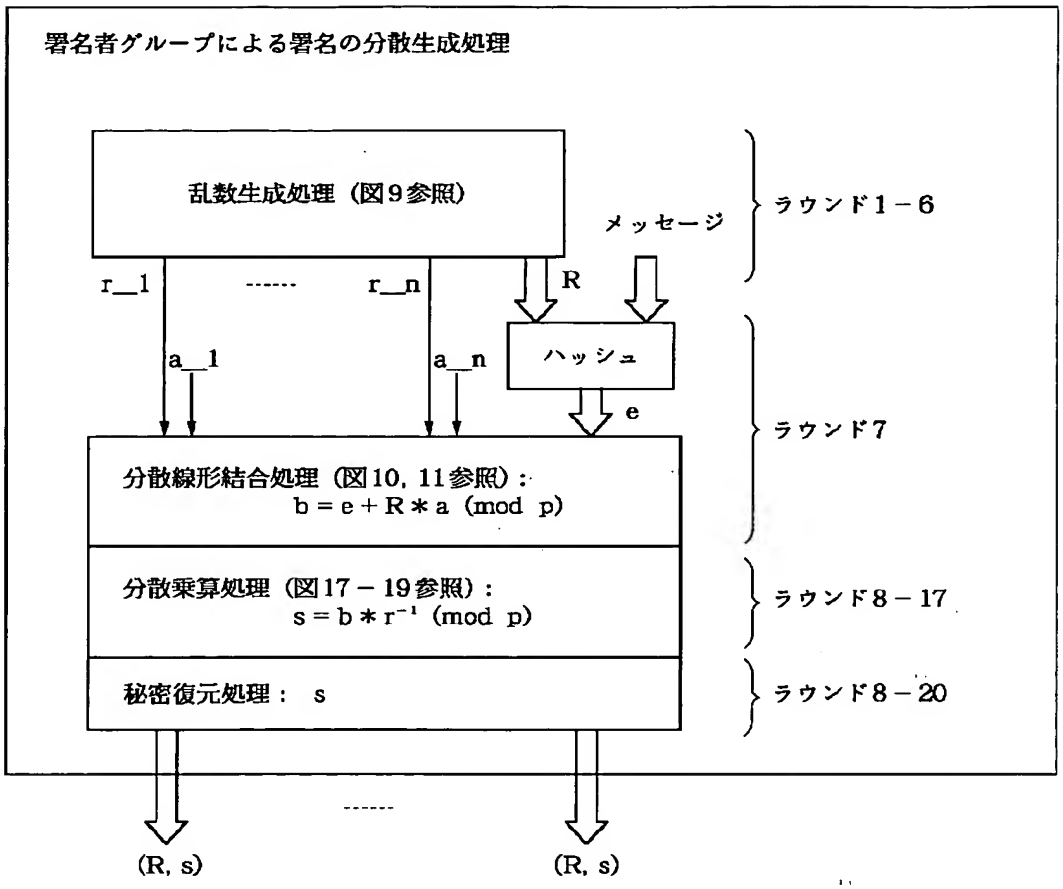
図17参照
加入者1 処理R6.1
加入者d 処理R6.d
加入者n 処理R6.n
各加入者iによって放送された情報 B5.i
加入者1 処理R7.1
加入者d 処理R7.d
加入者n 処理R7.n
各加入者iによって放送された情報 B6.i
加入者1 処理R8.1
加入者d 処理R8.d
加入者n 処理R8.n
各加入者iによって放送された情報 B7.i
加入者1 処理R9.1
加入者d 処理R9.d
加入者n 処理R9.n
各加入者iによって放送された情報 B8.i
加入者1 処理R10.1
加入者d 処理R10.d
加入者n 処理R10.n
各加入者oによって放送された情報 B9.o
加入者1 処理P1
加入者d 処理Pd
加入者n 処理Pn
各加入者iによって放送された情報 B10.i
各加入者iの出力s_i及び(s*y)_i

【図１９】


y_1
y_i
y_n
x_1
x_i
x_n
1
i
n
Z = X * Y
1
i
n
z_1
z_i
z_n


y_1
y_i
y_n
x_1
x_i
x_n
1
i
n
r(i)
確認可能な秘密分散及び分散乗算処理
r(1)+…+r(n)
r(1)y+…+r(n)y
r_i
(ry)_i
u=x−r
秘密復元処理
u
u
u
z=ry−uy
z_1
z_i
z_n
ラウンド1－10
ラウンド11
ラウンド12
ラウンド13－15
ラウンド16

【図２０】


署名者グループによる署名の分散生成処理
乱数生成処理（図９参照）
メッセージ
ラウンド１－６
r_1
------
r_n
R
a_1
a_n
ハッシュ
e
ラウンド７
分散線形結合処理（図10，11参照）：
b = e + R * a (mod p)
分散乗算処理（図17－19参照）：
s = b * r^-1 (mod p)
ラウンド８－17
秘密復元処理： s
ラウンド８－20

(R, s)
(R, s)

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-204697
(43)Date of publication of application : 09.08.1996

(51)Int.CI. H04L 9/00
H04L 9/10
H04L 9/12
G09C 1/00

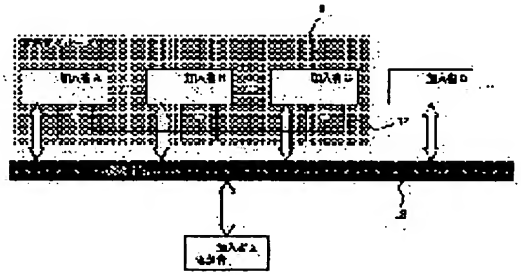
(21)Application number : 07-008185
(22)Date of filing : 23.01.1995

(71)Applicant : CANON INC
(72)Inventor : MANUERU SERESEDO
IWAMURA KEIICHI

(54) SIGNATURE GENERATING METHOD IN COMMUNICATION SYSTEM HAVING PLURAL EQUIPMENTS

(57)Abstract:
PURPOSE: To generate a signature distributingly with practical calculation amount and communication amount.

CONSTITUTION: In the communication system in which plural equipments are connected by a secret communication channel 13 used for information communication with secrecy to other equipments and a broadcast channel 12 sending information in common from each equipment to all other equipments, each equipment in a signature group selects 1st secrecy information at random and distributes the information with secrecy to each equipment in the group and each equipment applies 1st prescribed function on the 1st secrecy information. The obtained output is broadcast to all the equipments in the group, the 1st secrecy information is added and output values of each equipment are multiplied. Then the result of multiplication and the message are processed by using prescribed 2nd function and 2nd secrecy information is distributed by using the processing result, the result of sum and a public key to decode the distributed 2nd secret information and the decoded information is outputted as a signature together with the distribution multiplication result.



LEGAL STATUS

[Date of request for examination] 23.01.2002

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]
[Date of registration]
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]
[Date of extinction of right]